5-11-2013

Effect of Positional Dependence in Recognizing Transcription Factor Binding Sites

Saad A. Quader
University of Connecticut - Storrs, saad.quader@uconn.edu

Effect of Positional Dependence in Recognizing Transcription Factor Binding Sites

Saad Altaful Quader

B.Sc., Bangladesh University of Engineering & Technology, 2007

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

at the

University of Connecticut

2013

APPROVAL PAGE

Master of Science Thesis

Effect of Positional Dependence in Recognizing Transcription Factor Binding Sites

Presented by

Saad Altaful Quader, B.Sc.

Major Advisor _____
Chun-Hsi Huang

Associate Advisor _____
Sanguthevar Rajasekaran

Associate Advisor _____
Daniel Schwartz

University of Connecticut
2013

Acknowledgments

I am grateful to my mentor and advisor, Professor Dr. Chun-Hsi Huang, for this unending kindness, patience, generosity, support, and encouragement.

I am grateful to Professor Dr. Sanguthevar Rajasekaran for inspiring me in learning mathematics and computer science.

I am grateful to Professor Dr. Daniel Schwartz, who has kindly mentored me in many ways: not only with suggestions about my research but also with generous support for the most part of my stay at the University.

I am thankful to both Dr. Rajasekaran and Dr. Schwartz for their kind consent in being members of my masters advising committee. I am also thankful to the Department of CSE for their continuous support in the form of teaching assistantship.

At last but not the least, this research was supported in part by the National Science Foundation (US) under the grant CCF-0755373.

Contents

List of Tables

List of Figures

List of Abbreviations

AUC	Area Under Curve
BS	Binding Site
FP	False Positive
FPR	False Positive Rate
FN	False Negative
IC	Information Content
PS	Pairwise Score
PWM	Positional Weight Matrix
ROC	Receiver Operating Characteristics
TF	Transcription Factor
TFBS	Transcription Factor Binding Site
TN	True Negative
TP	True Positive
TPR	True Positive Rate

Abstract

Background: Many consensus-based and Position Weight Matrix-based methods for recognizing transcription factor binding sites are not well suited to the variability in the lengths of binding sites. Besides, many methods discard known binding sites while building the model. Moreover, the impact of Information Content (IC), and the positional dependence of nucleotides within an aligned set of TFBSs has not been well researched for modeling variable-length binding sites. In this paper, we propose *ML-Consensus*, a consensus model for variable-length binding sites which does not exclude any input binding sites. We consider Pairwise Score (PS) as a measure of positional dependence of nucleotides within an alignment of binding sites. We investigate how the prediction accuracy of ML-Consensus is affected by using IC, PS, and any particular binding site alignment strategy. We perform leave-one-out cross-validations on datasets of six species from the TRANSFAC public database, and analyze the results using ROC curves and Wilcoxon matched-pair signed-ranks test.

Results: We observed that the incorporation of IC and PS in ML-Consensus results in statistically significant improvement in the prediction accuracy. Moreover, any two positions in the multiple sequence alignment of the binding sites were found to be interdependent only when they the distance between them was below a certain value. Lastly, configurations with state-of-the-art alignment strategies did not perform significantly better than configurations with a naïve alignment strategy.

Conclusions: There exists a core region within a set of known binding sites,

and positions in that core region are interdependent. Additionally, it is possible to improve the existing state-of-the-art multiple sequence alignment algorithms by using such information as mentioned above about the core region among the binding sites.

Availability: All source codes (C#), results, supporting evidence, supplementary data and figures are available from
http://biogrid.engr.uconn.edu/mlconsensus/ .

1

Introduction

Background

Transcription factors (TF) are proteins that facilitate/repress the transcription process by binding to specific locations of the DNA. These locations are referred to as the *binding sites* (BS). In many cases binding sites of a transcription factor contain a common nucleotide pattern referred to as the *sequence motif* [1]. DNA motif-finding algorithms use various models to represent this motif. One of these models is the *consensus*, a sequence representation derived from a multiple sequence alignment of binding sites [2, 3]. The consensus sequence retains only the most conserved base at each position. However, this results in loss of information about other bases at that position [4]. *Position weight matrix* or PWM is another representation model which records frequency (or probability) of every base at each position of the multiple sequence alignment [1,5,6]. PWM is also called the *probabilistic sequence model* or *scoring matrix*. Scoring matrices are widely used by motif-finding tools for recognizing transcription factor binding sites [7–13]. These algorithms use mostly two techniques to generate the multiple sequence alignment for computing the PWM: (1) Expectation Maximization (EM) [9,10] or (2) variants of Gibbs Sampling ([11,12,14]). There

are other tools which use completely different representation and algorithm. For example, machine learning techniques like genetic algorithm (FMGA [15]), biclustering (MUSA [16]), ensemble (Yanover, [17]); graph algorithms (Winnower [18]), dictionary/suffix tree algorithms, dynamic programming, etc. Since motif-finding can be viewed as a generalizing problem, models for representing the motif, by nature, discard information specific to individual samples. The tool SiTaR [19] notices this issue and it does not make any generalization from the known TFBSs while building its model. Rather, it uses distance-metrics which are functions of each known TFBS. The survey article by Das and Dai [20] provides a classification of DNA motif-finding methods based on different representation models. Some tools can discover sequence motifs from a set of input sequences without using prior knowledge (called *de novo* or *unsupervised* motif finding), while some tools use a set of known TFBSs to develop the representation model and use it to detect potential binding sites, which can be called *supervised* motif detection. In the following discussion, we consider only supervised motif detection tools which use either PWM or consensus as the representation model.

Tools for TFBS detection may apply various constraints and restrictions on the input sequences. Although a PWM or scoring matrix has a fixed width, some TFBSs datasets show no variability in lengths of binding sites (e.g., the bacterial dataset described in [21]) whereas some datasets show remarkable variability in sequence lengths. In order to circumvent this variability, many tools (e.g., PMATCH [12]) align the input TFBSs and then discard the flanking regions, keeping a fixed-width portion from the middle of the original multiple sequence alignment from which the scoring matrix is computed. Other tools (e.g., MatInspector [8]) apply constraints and assumptions on the nature of binding sites, such as only fixed-length sites are considered, or only sites containing a fixed-length subsequence are considered. It is not yet confirmed, however, whether the protein-DNA binding mechanism indeed

follows such constraints. Some tools, e.g. PMATCH, excludes some documented binding sites based on constraints on the lengths of the sites, and imposes a constraint on the core region. (PMATCH defines the core region as the five most conserved positions within the alignment.) Then there is a question of allowing gaps in the representation model. Studies have found that the transcription factor p53 and different Stat family proteins bind with variable length sites which have variable-length gaps in their core regions [22–24]. However, some tools allow gaps in the PWM [25], while some do not allow any gaps [8, 14].

Although TFBS detection tools do not explicitly use an external multiple sequence alignment (MSA) tool/algorithm [20], an MSA algorithm is implicitly associated with any tool that generates a PWM/consensus. For example, PMATCH [12] uses Gibbs Sampling to align the given set of TFBSs. The MSA algorithm associated with such a model influences its performance because the alignment (and therefore, the scoring matrix or consensus) generated by different MSA algorithms will be different. A discussion and survey on the state-of-the-art MSA algorithms can be found in [26].

Basic consensus-based and PWM-based models assume that positions in a binding site are independent. However, some biological studies have suggested that the positions in a binding site are correlated [27, 28]. Several computational models for this correlation have been proposed [29, 30]. Some studies have described *pairwise score* (PS) or pairwise correlation score, a heuristic method that computes interdependence of any two positions in a set of aligned TFBSs [31]. These two positions are located within a fixed distance from one another; this distance is called the *scope* of PS. It should be noted that pairwise correlation score is not the same as the statistical measure "correlation"; rather, it is a measure of co-occurrence of bases within a given proximity (i.e., scope).It has been shown that using PS in the scoring function of basic consensus-based and PWM-based models results in statistically significant improvement in performance [31].

Information content (IC) of an alignment of binding sites is a measure of conservation of any base at any given position in that alignment. It has been shown that the addition of IC in the scoring function of basic consensus-based and PWM-based models results in statistically significant improvement in performance [2, 31].

Our Research

The restrictions imposed by many tools on the known TFBS sequences leave many known binding sites from consideration while building the representation model. Additionally, the study on the effect of IC and PS in the consensus/PWM model [31] was performed on datasets which had no variability in the lengths of binding sites [21]. Moreover, to the best of our knowledge, no study has been performed on how the range of positional interdependence affects the prediction accuracy of the model with pairwise score. Lastly, as far as we know, there is no study showing how the accuracy of the model is impacted with the choice of the underlying multiple sequence alignment algorithm.

Therefore, the aim of our study was to develop a consensus-based TFBSs representation model that would impose no restriction on the set of known TFBSs. Our model is called the (*Mixed-length Consensus* or *ML-Consensus*). This model was used to evaluate the impact of IC, PS, and alignment strategy on the prediction accuracy of the model.

Our study was made up of leave-one-out cross-validation experiments for training/testing our model on TFBS data for six species extracted from TRANSFAC public database [32]. All possible combinations of model parameters were considered: (1) with and without IC (2) without PS and with PS at different scopes (3) six different alignment strategies. The statistical evaluation of the performance data were performed through ROC curves and the Wilcoxon matched-pair signed-ranks test.

Specific Goals

In general, our goal was to study how the performance of the ML-Consensus model (see Section 2.3) changes with the change in the model parameters. These parameters are information content (Section 2.1), pairwise score and PS scope (Section 2.1), and lastly alignment strategy (Section 2.4). In particular, we had the following goals:

Goal 1. Measure how the performance of a model varies with the change in PS scope value. The motivation for this goal is to discover the degree of positional dependence within the core region shared by a set of TFBSs. A consensus or PWM alone does not contain any information about whether two different positions of the alignment matrix, when conserved, tend to be within a certain proximity. By definition, pairwise score rewards positional co-occurrence only within a certain scope. Thus consistent high performance of models with certain PS scopes may indicate information about the core region.

The hypothesis (based on [31]) was that using PS in scoring should improve the performance over a model that does not use PS, other model parameters being the same. Moreover, a model with a large PS scope should perform better than models with smaller PS scopes because, according to Equation (2.10), a score at any PS scope contains all information gathered in all smaller scopes.

Goal 2. Measure how models with different choices for state-of-the-art alignment strategies perform in comparison with a model using the naïve alignment algorithm (see Section 2.4), other model parameters remaining the same. The motivation for this goal was the following. General-purpose multiple sequence alignment tools perform various sophisticated computational measures in aligning a set of sequences. However, they do not make any special assumptions about aligning TFBSs. If ML-Consensus configurations with a state-of-the-art alignment strategy did not significantly outperform configurations using a naïve alignment strategy (other parameters remaining the same), it would indicate that there is some information which the

state-of-the-art alignment algorithm could use in order to improve its performance for aligning TFBSs in general.

The hypothesis was that models with state-of-the-art alignment strategies should perform significantly better than a model with naïve alignment strategy. The basis of the hypothesis is that the naïve alignment algorithm is simplistic and heuristic, while all other alignment strategies are state-of-the-art.

Goal 3. Measure how the usage of IC to a model not using IC affects the performance. The motivation was the following. That IC improves performance of consensus model over fixed-length TFBSs is already known [31]. However, similar studies which cover variable-length TFBSs are not available. The hypothesis was that incorporating IC would improve the performance of the ML-Consensus model, other parameters remaining the same.

Main Results

Our results showed that the adoption of IC or PS in the scoring function of ML-Consensus resulted in significant improvement in performance. Moreover, it was found that a large PS scope (e.g., the full scope) did not produce the best performance for a given configuration; performance of the model decreased after PS scope grew larger than a certain value. Additionally, it was also found that the models with state-of-the-art alignment algorithms did not produce the best performance.

These results indicated a way to estimate the minimum length of the core region from a set of known TFBSs. These results also suggested that existing state-of-the-art MSA algorithms could be improved by means of utilizing prior information and assumptions about TFBS. However, such an algorithm is a matter of further investigation.

Part of the work and results presented in this thesis appeared at the proceedings

of the EvoBio'11[1] conference [33] and at the journal BMC Research Notes [34].

The rest of this article is organized as follows. Our model and methods are presented in Chapter 2. The results are presented in Chapter 3. The discussions are presented in Chapter 4.

[1] 9th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Computational Biology

2

Model and Methods

In this section, we start with presenting the mathematical definition of the ML-Consensus model and its various parts. Next we describe how we collected and processed the input data to build training and testing datasets. Then we describe how we made statistical evaluation of the experiments through ROC curves and Wilcoxon matched-pair signed-ranks test.

2.1 Definitions

Consensus

Let S be the set of N binding sites for a particular transcription factor. Let A be a multiple sequence alignment matrix of S with width of M. A gap in alignments in A is denoted by '-'.

Let $n_j(b)$ be the number of times base $b \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$ appears at j^{th} position of A. Let $f_j(b) = n_j(b)/N$ be the corresponding frequency. Similarly, let $n(b)$ be the number of times base b appears overall in A, and $f(b)$ be the overall frequency for base b in A.

b	A	A	A	C	C	G
d	C	G	T	G	T	T
$amb(b,d)$	I	J	K	L	M	N

Table 2.1: Ambiguity codes.

A letter representing more than one nucleotides is called the *ambiguity code* for those nucleotides [35]. Let $amb(b,d)$ be the ambiguity code for two bases $b,d \in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\}$ as described in Table 2.1, and $amb(b,*)$ be any ambiguity code involving base b.

Let C be the consensus sequence derived from A, and $C_j, 1 \leq j \leq M$ be the j^{th} letter in C. Then, the consensus is computed as follows:

$$C_j = \begin{cases} b & f_j(b) > 0.5, \forall b \in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\} \\ amb(b,d) & f_j(b) + f_j(d) > 0.75, \forall b,d \in \{\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}\} \\ '-' & \text{Otherwise} \end{cases} . \tag{2.1}$$

Overlap of Two Sequences, w

Let s_1 and s_2 be two sequences of length l_1 and l_2, respectively, where $l_1 > 0$ and $l_2 > 0$. There can be $l_1 + l_2 - 1$ different ways of placing these two sequences side-by-side such at each case, the beginning of s_1 is aligned with a different position in s_2. Each of these l possible alignments of s_1 and s_2 is called an overlap, and is denoted by w. At each overlap, a sequence of blank characters, denoted by -, are padded at both ends of these sequences so that they are augmented to have the same length. The set of all overlaps is denoted by W. It follows that $|W| = l_1 + l_2 - 1$. Table 2.2 shows all overlaps between the sequences "ABCDE" and "1234". There are $5 + 4 - 1 = 8$ overlaps.

Each overlap $w_k \in W, 1 \leq k \leq l_1 + l_2 - 1$ is a 3-tuple (k, l_1, l_2). The length of the overlap w_k is given by the following function

ABCDE--- ----1234	ABCDE 1234-
ABCDE-- ---1234	-ABCDE 1234--
ABCDE- --1234	--ABCDE 1234---
ABCDE -1234	---ABCDE 1234----

Table 2.2: All overlaps of the sequences "ABCDE" and "1234".

$$length(w_k) = \begin{cases} l_1 + l_2 - k & 1 \leq k < min(l_1, l_2) \\ max(l_1, l_2) & min(l_1, l_2) \leq k \leq max(l_1, l_2) \\ max(l_1, l_2) + k - min(l_1, l_2) & max(l_1, l_2) < k \leq l_1 + l_2 - 1 \end{cases} \quad (2.2)$$

The original sequences $s_i, i \in \{1, 2\}$ are augmented in the overlap w_k by adding zero or more blank characters at each side so that they both have the same length. Let $repeat(c, n)$ be the sequence made up of n occurrences of the character c. Let $concat(s_1, s_2, \ldots, s_n)$ denote the concatenation of sequences $s_1, s_2, \ldots, s_n$. Let us define the padding function as follows which adds a n_1 occurrences of the character c at the left, and n_2 occurrences of the same character at the right, of the sequence s.

$$pad(s, c, n_1, n_2) = concat\left(repeat\left(c, n_1\right), s, repeat\left(c, n_2\right)\right) \quad (2.3)$$

Then, the augmented sequences at overlap w_k are given by the following function:

$$f_{aug}(w_k, s_i) = padding\left(s_i, '-', \frac{\lfloor length(w_k) - l_i \rfloor}{2}, \frac{\lceil length(w_k) - l_i \rceil}{2}\right), \quad (2.4)$$

where $k \in \{1, 2\}$ and l_i is the length of s_i.

The Scoring Function, $\sigma(t, C)$

Let t be a putative binding site and C be the consensus from the alignment matrix A. Let t_j be the j^{th} base of t. The scoring function σ takes t and C as input, and computes the score of t with respect to C. This score, a real number, is an estimate of the similarity between t and C.

Let W be the set of all overlaps between t and C (see Section 2.1). For each overlap $w \in W$, let $C_{w,i}$ be the base in consensus corresponding to the i^{th} position in w. Define $t_{w,i}$ in similar way. For each overlap w, let $\sigma(t, C, w)$ be the score of t at that particular overlap; this score is equal to the number of matches between t and C at w:

$$\sigma(t, C, w) = \sum_{i \in w} Match(w, i) \ , \tag{2.5}$$

where

$$Match(w, i) = \begin{cases} 1 & : C_{w,i} = t_{w,i} \\ 1 & : C_{w,i} = amb(t_{w,i}, *) \\ 0 & : otherwise \end{cases} \ . \tag{2.6}$$

Finally, $\sigma(t, C)$ is the maximum score across all overlaps w.

$$\sigma(t, C) = \max_{w} \left(\sigma \left(t, C, w \right) \right) \ . \tag{2.7}$$

Computing $Match(w, i)$ takes $O(1)$ time, and computing $\sigma(t, C, w)$ takes $O(M)$ since size of w is $O(M)$. Finally, the score of t with respect to C is the maximum score obtained in all overlaps, which takes $O(M^2)$ since there can be at most $O(M)$ overlaps.

Pairwise Score, PS

Pairwise score is a measure of interdependence among positions in a binding site with respect to the consensus [31]. Two different positions in an overlap are consid-

Sequence 1	-ACATATGG
Sequence 2	GATATCGG-
Matches	.*.**..*.
Positions	123456789
Pairwise matches at Scope 1	(4,5)
Pairwise matches at Scope 2	(4,5), (2,4)
Pairwise matches at Scope 3	(4,5), (2,4), (2,5), (5,8)
Pairwise matches at Scope 4	(4,5), (2,4), (2,5), (5,8), (4,8)
Pairwise matches at Scope 5	(4,5), (2,4), (2,5), (5,8), (4,8)
Pairwise matches at Scope 6	(4,5), (2,4), (2,5), (5,8), (4,8), (2,8)
Pairwise matches at Full Scope	(4,5), (2,4), (2,5), (5,8), (4,8), (2,8)

Table 2.3: Pairwise dependence between two sequences in an overlap. Positions 2, 4, 5, and 8 have matches.

ered interdependent if there are matches in both positions in the alignment. Such interdependent positions are called a pairwise match. Number of pairwise matches in an overlap w can be used to improve the scoring function in Equation (2.5) which does not account for positional co-occurrence.

The *scope* of the pairwise interdependence is a parameter used in pairwise scoring which requires the members of a position-pair match to be close together. Formally, the *PS scope*, $K \geq 1$, is the maximum distance allowed between the members of a positions-pair match. Equivalently, position-pairs that are separated by more than K positions are not considered in pairwise scoring with scope K.

A special case of PS scope is *full PS scope*, denoted by $K = \infty$, which implies that the scope in Equation (2.10) will span the entire length of the overlap w (that is, $K = \infty \Rightarrow K = length(w)$) so that all position-pairs within the overlap are examined. Moreover, $K > length(w)$ is not meaningful. These scenarios are consolidated by the following function:

$$K_w(K, w) = \min\left(K, length(w)\right), K = 1, 2, 3, \cdots, \infty \tag{2.8}$$

Table 2.3 shows an example of pairwise interdependence in different scopes.

Now we shall define the scoring function which incorporates pairwise score. Let positions i and $i + k$ be separated by k positions in overlap w where $k \geq 1$ and

$i + k \leq length(w)$. The match-score for this position-pair, $MatchPair(w, i, k)$, is defined as follows:

$$MatchPair(w, i, k) = \begin{cases} 2 & : Match(w, i) = 1 \\ & \text{and} \\ & Match(w, i + k) = 1 \\ 0 & : otherwise \end{cases}, \tag{2.9}$$

where $1 \leq k \leq length(w) - i$. Computing this function takes $O(1)$ time. Now let K be the scope for pairwise scoring. Let $length(w)$ be the length of the overlap. The pairwise score of t at overlap w, $\sigma_{\mathrm{PS}}(t, C, w)$, is defined as the total number of position-pair matches for all positions situated within the scope of PS.

$$\sigma_{\mathrm{PS}}(t, C, K, w) = \sum_{s=1}^{\min(K, length(w))} \sum_{i=1}^{length(w)-s} MatchPair(w, i, s) \tag{2.10}$$

When not using full scope, $K = O(1)$ and this operation takes $O(MK^2)$ time. Otherwise, $K = length(w) = O(M)$, and this operation takes $O(M^3)$.

Now, Equation (2.10) can be rewritten as

$$\begin{aligned} \sigma_{\mathrm{PS}}(t, C, K, w) &= \sum_{s=1}^{K-1} \sigma_{\mathrm{PS}}(t, C, s, w) + \\ & \qquad \sum_{i=1}^{length(w)-K} MatchPair(w, i, s)\,, \end{aligned} \tag{2.11}$$

for $K \geq 1$. From this we can see that the pairwise score at larger scopes can be computed in bottom-up fashion from scores at smaller scopes.

Finally, $\sigma(t, C, K)_{\mathrm{PS}}$ is the maximum score across all overlaps w.

$$\sigma_{\mathrm{PS}}(t, C, K) = \max_{w} \left(\sigma_{\mathrm{PS}}\left(t, C, K, w\right)\right)\,. \tag{2.12}$$

Putting $K = \infty$ will compute pairwise score with full scope.

Information Content, IC

Information Content (also called *entropy*) at any position j of the alignment A is a measure of conservation of any base at that position [5, 36]. If a base is highly conserved at a position, chance of encountering a different base at that position is small; thus the information content at that position is low. The IC at position j of the alignment matrix A is defined as:

$$IC(A, j) = 2 + \sum_{b \in \{\text{A,C,G,T}\}} f_j(b) \log f_j(b) \ , \tag{2.13}$$

where the term $f_j(b) \log f_j(b)$ becomes zero whenever $f_j(b)$ becomes zero, thus avoiding evaluation of $\log 0$. $IC(A, j)$ for all j can be computed in $O(M)$ time.

Scoring Function with Information Content, $\sigma_{\text{IC}}(t, C)$

Let $A(w, i)$ be the position in A that corresponds to the i-th position in w. When information content (IC) (see Section 2.1)is used in scoring, the scoring function for the overlap becomes:

$$\sigma_{\text{IC}}(t, C, w) = \sum_{i \in w} Match(w, i) \cdot IC\left(A, A\left(w, i\right)\right) \ . \tag{2.14}$$

This takes $O(M)$ time when $IC(A, j)$ are pre-computed.

Finally, $\sigma_{\text{IC}}(t, c)$ is the maximum score across all overlaps w.

$$\sigma_{\text{IC}}(t, C) = \max_{w} \left(\sigma_{\text{IC}}\left(t, C, w\right)\right) \ . \tag{2.15}$$

Scoring Function with both Information Content and Pairwise Score, $\sigma_{\text{ICPS}}(t, C, K)$

At any overlap w, let $n_{ij}(b, d)$ be the number of times two bases b and d appear together at positions i and j, respectively. Let $f_{ij}(b, d) = n_{ij}(b, d)/N$ be the cor-

responding frequency. Then, IC of position-pair (i, j) in the alignment matrix A is defined as follows:

$$IC_{\text{pair}}(A, i, j) = 4 + \sum_{b,d \in \{\text{A,C,G,T}\}} f_{ij}(b, d) \log f_{ij}(b, d) \tag{2.16}$$

Computing $f_{ij}(b, d)$ for all i, j, b, d takes $O(M^2)$ time. After that, computing $IC_{\text{pair}}(A, i, j)$ for all i, j takes $O(M^2)$ time.

Let $A(w, i)$ be the position in A that corresponds to the i-th position in w. Let $IC_{\text{pair}}(w, i, k)$ be the information content of the position-pair $(i, i + k)$ in w, which is defined as follows:

$$IC_{\text{pair}}(w, i, k) = IC_{\text{pair}}\left(A, A\left(w, i\right), A\left(w, i + k\right)\right) \tag{2.17}$$

Finally, the score of t at overlap w is defined as follows:

$$\sigma_{\text{ICPS}}(t, C, K, w) = \tag{2.18}$$
$$\sum_{s=1}^{min(K, length(w))} \sum_{i=1}^{length(w)-s} MatchPair(w, i, s) \cdot IC_{\text{pair}}(w, i, s)\,.$$

This takes $O(MK^2)$ time because all $IC_{\text{pair}}(A, i, j)$ values are already computed for all i, j.

Finally, $\sigma_{\text{ICPS}}(t, C, K)$ is the maximum score across all overlaps w.

$$\sigma_{\text{ICPS}}(t, C, K) = \max_{w} \left(\sigma_{\text{ICPS}}\left(t, C, K, w\right)\right) \quad . \tag{2.19}$$

2.2 Problem Description

The general supervised TFBS detection problem can be described as follows:

Problem 1 (Supervised TFBS Detection Problem)**.** *Let S be a set of known TFBSs of the TF T, and t be any sequence. Let σ be a scoring function which maps an arbitrary sequence to its similarity score $\sigma(t, S)$ such that sequences similar to those in S will have high scores. Let θ be a suitable cut-off score. Then, if $\sigma(t) \geq \theta$, return* `True` *(which means t is a potential binding site of T). Otherwise, return* `False`*.*

PMATCH [12] and JASPAR [13] are examples of TFBS recognition tools that consider the above problem.

The ML-Consensus model, however, considers the following weaker version of the above problem.

Problem 2 (Weak Supervised TFBS Detection Problem)**.** *Let S be a set of all known TFBSs of the TF T, and $p \in S$ be another known binding site of T. Let t be any sequence. Let $S^prime = S \setminus \{p\}$. Let σ be a scoring function which maps an arbitrary sequence to its similarity score $\sigma(t, S^prime)$ such that sequences similar to those in S will have high scores. Then, if $\sigma(t, S^prime) \geq \sigma(p, S^prime)$, return* `True` *(which means t is a potential binding site of T). Otherwise, return* `False`*.*

The assumption behind this problem definition is that all binding sites of T must share some similarity, and any true binding site of T should score higher than any sequence that is not a binding site of T. We considered the weaker Problem 2 because our goal at this stage of the research was not to solve the supervised TFBS detection problem in its generality. Rather, our goal was to study how various model parameters (see Section 1) affects the overall quality of the results. Moreover, computing the optimal cut-off score for a given set of TFBS is a non-trivial problem in itself and is not relevant to the goal stated above. We consider this as a future extension of this research.

2.3 ML-Consensus Model

Below we specify the parameters, components, input, and output of the ML-Consensus model in solving the Problem 2.

Input

1. T, a transcription factor.
2. S, the set of all known TFBSs for T. S must have at least 3 binding sites.
3. p, a known BS of T such that $p \in S$.
4. t, an unknown sequence.

ML-Consensus Model Parameters

1. *IC:* (Either `True` or `False`.) Denotes whether the model should consider information content in scoring function.
2. *PS:* (Either `True` or `False`.) Denotes whether the model should consider pairwise score in scoring function.
3. *K:* (Non-negative integer.) The scope for pairwise score, valid when the parameter PS is set to `TRUE`. If $K = 0$, it implies that PS is `False`.
4. M: A multiple sequence alignment algorithm according to other model parameters. Input to this algorithm will be $S \setminus \{p\}$, that is, all sequences of S except p. The parameters $\{IC, PS, K\}$ determine the scoring function used in the naïve alignment algorithm described in Section 2.4, but do not have any effect on other alignment algorithms.

Components of ML-Consensus Model

1. A: The alignment matrix generated by the multiple sequence alignment algorithm M from the sequences $S \setminus \{p\}$.

2. C: The consensus computed from A.

3. $\sigma(t, C, IC, PS, K)$: A scoring function which computes the score of an arbitrary sequence t with respect to the consensus C. Choice of this function depends on the model parameters IC, PS, and K.

$$\sigma(t, C, IC, PS, K) = \tag{2.20}$$

$$\begin{cases} \text{Equation (2.7)} & \text{If not using IC or PS} \\ \text{Equation (2.15)} & \text{If using IC but not PS} \\ \text{Equation (2.12)} & \text{If using PS scope } K\text{, but not IC} \\ \text{Equation (2.19)} & \text{If using IC with PS scope } K \end{cases}$$

Output

- `True`, if $\sigma(t, C, IC, PS, K) \geq \sigma(p, C, IC, PS, K)$.
- `False`, otherwise.

Figure 2.1 shows the steps in such a leave-one-out experiment.

2.4 Alignment Strategies

The consensus (or PWM) of a set of TFBSs depends on the multiple sequence alignment algorithm used to create the alignment matrix from the give set of TFBSs. Therefore, the choice of alignment algorithm directly affects the effectiveness and performance of a TFBS recognition model that uses the consensus (or PWM) of the known set of TFBSs. A goal of our study (Goal 2, see Section 1) was to measure the change in performance of two models with different alignment strategies: one with a state-of-the-art multiple sequence alignment algorithm and the other with a naïve heuristic algorithm, all other variables remaining the same. Therefore we devised a heuristic multiple sequence alignment algorithm (called the *naïve* algorithm) and contrasted it with five other state-of-the-art multiple sequence alignment algorithms:

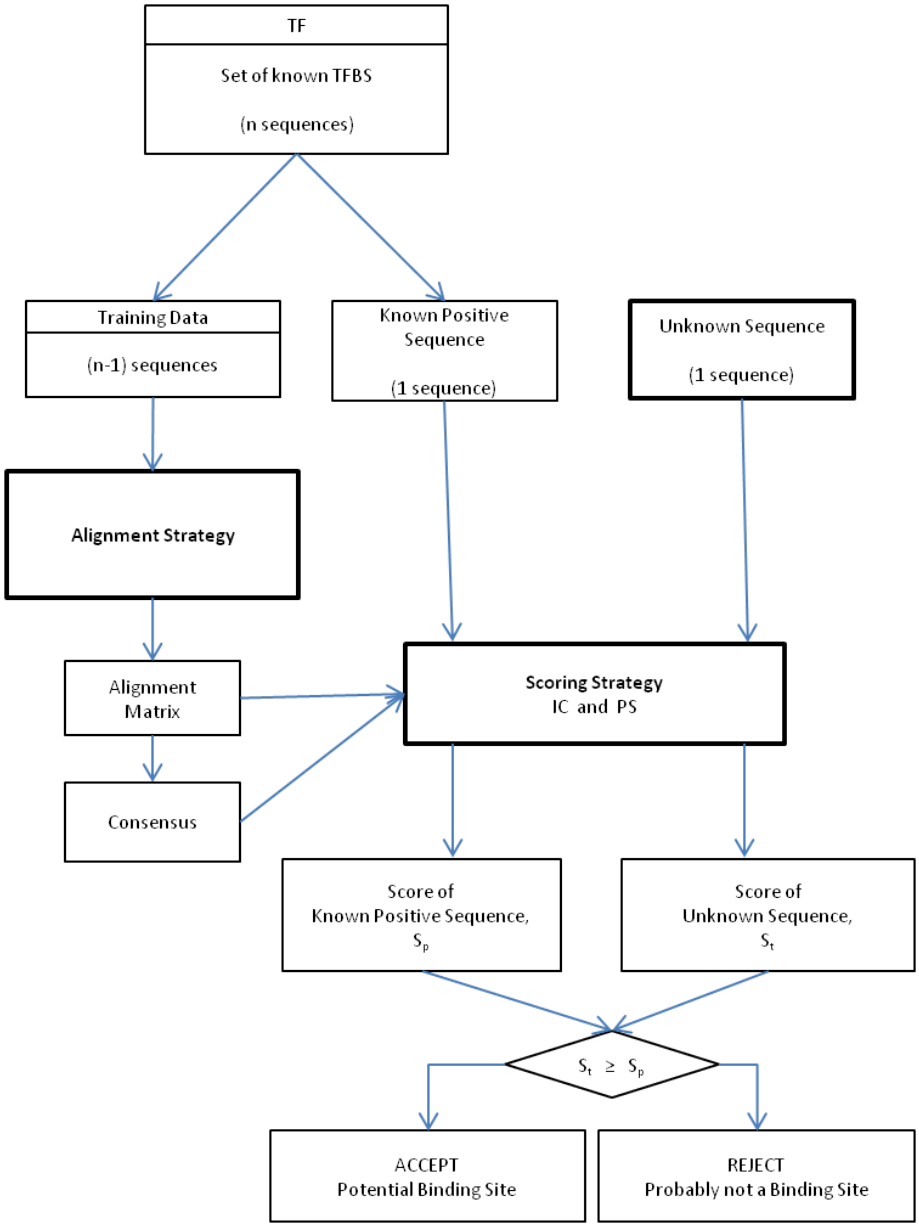

FIGURE 2.1: Given the set S of all known binding sites of the transcription factor T, and a specific binding site $p \in S$, the ML-Consensus model determines whether an unknown sequence t is a potential binding site of T if its score is greater than or equal to the score of p.

Original, unaligned	Naïve, no IC, no PS	Naïve, no IC, PS scope 4
GAGAAAAAGCCATTAGAG GCGTAATGTGTT GAGCAATTACAG TAGTAATAATG TAATTATTAAA	TAGTAATAATG---------- ---TAATTATTAAA------- ---GCGTAATGTGTT------ ---GAGCAATTACAG------ ---GAGAAAAAGCCATTAGAG ******* GAGTAAT	TAGTAATAATG-------- ---TAATTATTAAA----- ---GCGTAATGTGTT---- GAGCAATTACAG------- -GAGAAAAAGCCATTAGAG ****** AATAAT

Naïve, IC, no PS	Clustal	MAFFT
------TAGTAATAATG--- ----------TAATTATTAAA ------GCGTAATGTGTT-- ------GAGCAATTACAG-- GAGAAAAAGCCATTAGAG-- ******** AGTAATTA	GAGAAAAAGCCATTAGAG ------GCGTAATGTGTT ------GAGCAATTACAG ------TAGTAATAATG- ------TAATTATTAAA- ******** * AGTAATTA-A	GAGAAAAAGCCATTAGAG G------CGTAATGTGTT G------AGCAATTACAG T------AGTAATAATG- T------AATTATTAAA- * ******** * G------AGTAATTA-A

Muscle	ProbCons	T-Coffee
------TAATTATTAAA- ------TAGTAATAATG- ------GCGTAATGTGTT GAGAAAAAGCCATTAGAG ------GAGCAATTACAG ******** * AGTAATTA-A	GAGAAAAAGCCATTAGAG GCGT------AATGTGTT GAGC------AATTACAG TAGT------AAT-AATG TAAT------TAT-TAAA **** *** * ** GAGT------AAT-A-AG	GAGAAAAAGCCATTAGAG GCGTA------ATGTGTT GAGCA------ATTACAG TAGTA------ATAA-TG TAATT------ATTA-AA ***** **** ** GAGTA------ATTA-AG

Table 2.4: Alignment matrices from different alignment strategies. An asterisk (*) below each alignment matrix denotes a conserved position. The last line is the consensus computed by Equation 2.1. The naïve algorithm was used with information content but without pairwise score.

Clustal Omega [37], MAFFT [38], Muscle [39], ProbCons [40], and T-Coffee [41]. An overview of how these algorithms work can be found in [26]. Table 2.5 shows the versions and command line options of these tools. Table 2.4 shows the alignment matrices produced by different alignment strategies from the same input. Naturally, the consensus found from these alignment matrices are different.

The naïve multiple sequence alignment algorithm

The assumption behind this multiple sequence algorithm is that there is a core region which is shared by all binding sites. Therefore on average, positions in short sites are more likely to constitute the core region (than positions in long sites). The algorithm

Tool	Command Line Options	Default Iterations	Remarks
Clustal Omega v1.0.3 Ubuntu x86_64	–iter=10		
MAFFT v6.925b Windows 64 bit	–localpair –maxiterate 1000		L-INS-i
ProbCons v1.12 Ubuntu x86_64		-ir=100	
Muscle v3.8.31 Ubuntu x86_64		-maxiters=16	Fastest
T-Coffee v9.03.r1318 Ubuntu x86_64		-iterate=0	Slowest

Table 2.5: Version and command line options for different multiple sequence alignment tools.

is described in Algorithm 2.1. The same set of model parameters $\{IC, PS, K\}$ used in choosing the scoring function (see Section 2.3) must be used in parameterizing the naïve algorithm.

It can be observed that the order of choosing binding sites affects the resultant alignment. As an extreme example, consider a set of 10 binding sites where 7 of them are very similar to each other and three binding sites are very different in composition. If these odd sites happen to be the shortest ones, they are likely to negatively impact the rest of the alignment (because all other sites will be aligned with respect to these heterogeneous sites). However, according to the assumption of the algorithm, if the short sites contain the core region then they are less likely to be heterogeneous.

2.5 Leave One Out Experiments

Input

We extracted TFBS data from TRANSFAC public database [32]. We considered TFs with at least three binding sites. Table 2.6 shows basic statistics for this data.

Algorithm 2.1 The pseudo code of the naïve alignment algorithm.

Input: S, a set of sequences.
Input: A, an empty matrix.
Input: $IC \in \{0, 1\}$, denoting whether information content should be used in scoring.
Input: $PS \in \{0, 1\}$, denoting whether pairwise score should be used in scoring.
Input: K, the scope for pairwise score when $PS = 1$.
Input: $\sigma(t, C)$, a scoring function chosen from Equations (2.7), (2.12), (2.15), or (2.19), depending on parameters IC, PS, and K
Output: A, a matrix containing all sequences of S in aligned format.

```
n ← |S|                                              ▷ Number of sequences
m ← length of the longest sequence ∈ S
L ← 2 * m − 1                                        ▷ Width of alignment matrix
num_cols(A) ← L
num_rows(A) ← n
T ← sort(S) by length, from shortest to longest. Sequences of the same length are ordered at random.
for i = 1 : n do
    t ← T[i]                                         ▷ Pick the next shortest sequence
    l ← length(t)                                    ▷ Length of this sequence
    t* ←NULL                                         ▷ Sequence t in aligned position
    if i = 1 then                                    ▷ First sequence
        left ← sequence of ⌊(L − l)/2⌋ blank characters
        right ← sequence of ⌈(L − l)/2⌉ blank characters
        t* ← string_concat(left, t, right)           ▷ Now length(t*) = num_cols(A)
    else
        C ← consensus(A[1 : i − 1])                  ▷ Equation (2.1)
        W ← {(k, L, l)}                              ▷ all overlaps between C and t
        m ← 0                                        ▷ Maximum score of t across all overlaps
        for w_k ∈ W do
            C' = f_aug(w_k, C)                       ▷ Equation (2.4)
            t' = f_aug(w_k, t)
            s = σ(t', C')                            ▷ The score of t at overlap w_k
            if s > m then
                m ← s                                ▷ Maximum score so far
                t* ← t'                              ▷ Best alignment for t so far
            end if
        end for
    end if
    A[i] ← t*                                        ▷ Add t to alignment matrix
end for
```

Species	TF	BS	Average BS length	Standard Deviation
D. melanogaster	29	352	12.14	5.83
G. gallus	23	179	7.78	5.47
H. sapiens	179	2493	13.93	7.11
M. musculus	125	1266	10.13	6.01
R. norvegicus	59	795	13.47	6.80
S. cerevisiae	42	385	9.17	5.18
All species	457	5470	11.97	6.43

Table 2.6: Statistics of input TFBS data. The Standard Deviation (SD) column is the average of population SD of binding-site length for all TFs in the input dataset.

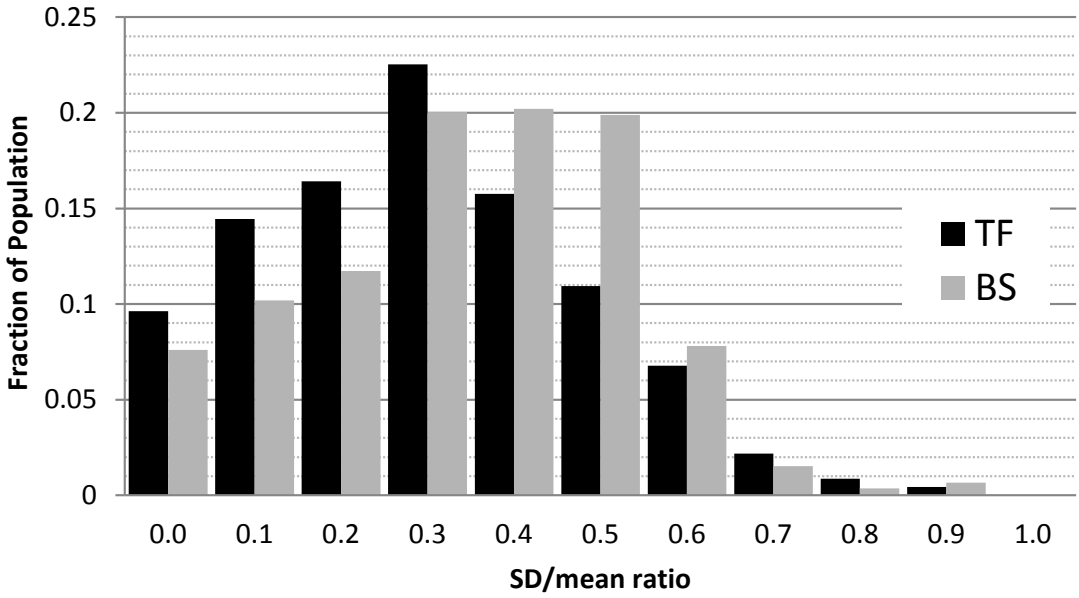

FIGURE 2.2: Histogram for variability in TFBS length in our input data from Transfac Public database. Population standard deviation and mean have been counted for each set of TFBS. The x-axis shows the ratio of SD and mean which can be seen as a measure of variability in data. Each TF and all its BSs are assigned into one bin in x-axis. The y-axis shows the fraction of total TF or BS population that fell into each bin in x-axis.

Figure 2.2 shows the variability in TFBS lengths in the input data. The x-axis shows the ratio of population SD and mean in BS length computed for a set of TFBS. From the figure it can be observed that 9.5% TFs have small deviation in size (the first bin of histogram) but they cover only 7.5% of total BSs. From first three bins, it can be seen that 40% of TFs (covering 29% BSs) have low variability ($\frac{\text{SD}}{\text{mean}} < 0.3$). From next three bins, it can be observed that another 49% TFs (covering 60% BSs) have much higher variability ($0.3 \leq \frac{\text{SD}}{\text{mean}} < 0.6$). Remaining 11% TFs have extreme variability, and they cover the remaining 11% of BSs.

For the sake of formalism, let us define $G = \{g_l, 1 \leq l \leq 6\}$ as the set of all species. Let N_{TF} be the total number of transcription factors in the dataset. Each transcription factor $T_i, 1 \leq i \leq N_{\text{TF}}$ is uniquely associated with a species $g(T_i) \in G$.

Experiment Configurations

As specified as the goals of this research (see Section 1), we studied the effect of three model parameters on the performance of ML-Consensus. These are (1) multiple sequence alignment strategy, (2) information content, and (3) pairwise score. There were six choice for alignment strategy (naïve, Clustal Omega, MAFFT, Muscle, ProbCons, and T-Coffee), two choices for IC (either using IC, or not), and twelve choices for PS (not using PS, PS scopes 1–10, full PS scope). We used 10 as the maximum PS scope (not considering full scope) because we had to have a finite number of choices for PS scope value. Thus in total there are $N_{config} = 6 \times 2 \times 12 = 144$ possible different parameter-combinations for ML-Consensus model. A model with a specific parameter-combination is called an *experiment configuration*, or *configuration* in short. Each of these configurations was trained and tested using the same input, training, and test data.

Let $\Theta = \{\theta_k, 1 \leq k \leq N_{config}\}$ be the set of all possible configurations, and let $\theta_k \in \Theta, 1 \leq k \leq N_{config}$ be the k^{th} configuration.

Training dataset for a leave-one-out experiment

Let $T_i, 1 \leq i \leq N_{TF}$ be the i^{th} transcription factor belonging to the species $g(T_i) \in G$. Let $S_i = \{s_j\}, 1 \leq j \leq N_i^{\text{BS}}$ be the set of all known binding sites for the TF T_i. For each sequence $s_j \in S_i$ we created a leave-one-out input dataset $S_i^j = S_i \setminus \{s_j\}$, which is the collection of all binding sites from S_i except the sequence s_j. Clearly, $|S_i^j| = |S_i| - 1$. Every training dataset S_i^j was written into a text file.

Test dataset for a leave-one-out experiment

Let X_i^{Neg} be the set of known negative examples for the i^{th} transcription factor T_i such that

$$X_i^{Neg} = \{s \in S_k, 1 \leq k \leq N_{\text{TF}} | k \neq i \vee g(T_i) = g(T_j) \vee s \notin S_i\}$$

. That is, X_i^{Neg} was made up of all binding sites of other TFs of the same species as T_i such that those sites were not a BS of T_i. The same set of known negative examples X_i^{Neg} was used as test data with all leave-one-out training datasets $S_i^j, 1 \leq j \leq N_i^{\text{BS}}$ corresponding to the TF T_i. However, each leave-one-out input dataset S_i^j had a different known positive example $s_j \in S_i$. Thus the combined test dataset for the leave-one-out input dataset S_i^j was $X_i^j = \cup\{s_j\}X_i^{Neg}$ which contained exactly one positive example (the first item in the set) and all known negative examples for T_i. Every test dataset X_i^j was written into a text file.

Building alignment matrices and scoring the test datasets.

Let $M(\theta_k)$ be the alignment algorithm of the configuration $\theta_k \in \Theta$. For each configuration $\theta_k \in \Theta$, $M(\theta_k)$ was used on every leave-one-out input dataset $S_i^j, 1 \leq i \leq N_{\text{TF}}, 1 \leq j \leq N_i^{\text{BS}}$ and the corresponding output, the multiple sequence alignment matrix $A_i^j(\theta_k)$, was written into text file. This matrix was used to build the consensus $C_i^j(\theta_k)$ which was used by the scoring function σ_{θ_k} to score every sequence in the test dataset X_i^j. The number of known negative examples which scored higher than the known positive example was recorded as the outcome of each leave-one-out experiment. Table 2.7 shows the outcome of the leave-one-out experiments conducted for the configuration $\theta = \{IC = \texttt{False}, PS = \texttt{False}, K = \texttt{Null}, M = \texttt{Clustal}\}$ over the first TF (named `dm001`) which belonged to the species *D. Melanogaster*. Outcomes of all leave-one-out experiments pertaining to each configuration $\theta_k \in \Theta$ was written to a single output file for θ_k. These output files were used in analyzing performance of individual configurations.

Figure 2.3 depicts the steps involved in generating the outcomes of all leave-one-out experiments belonging to each configuration.

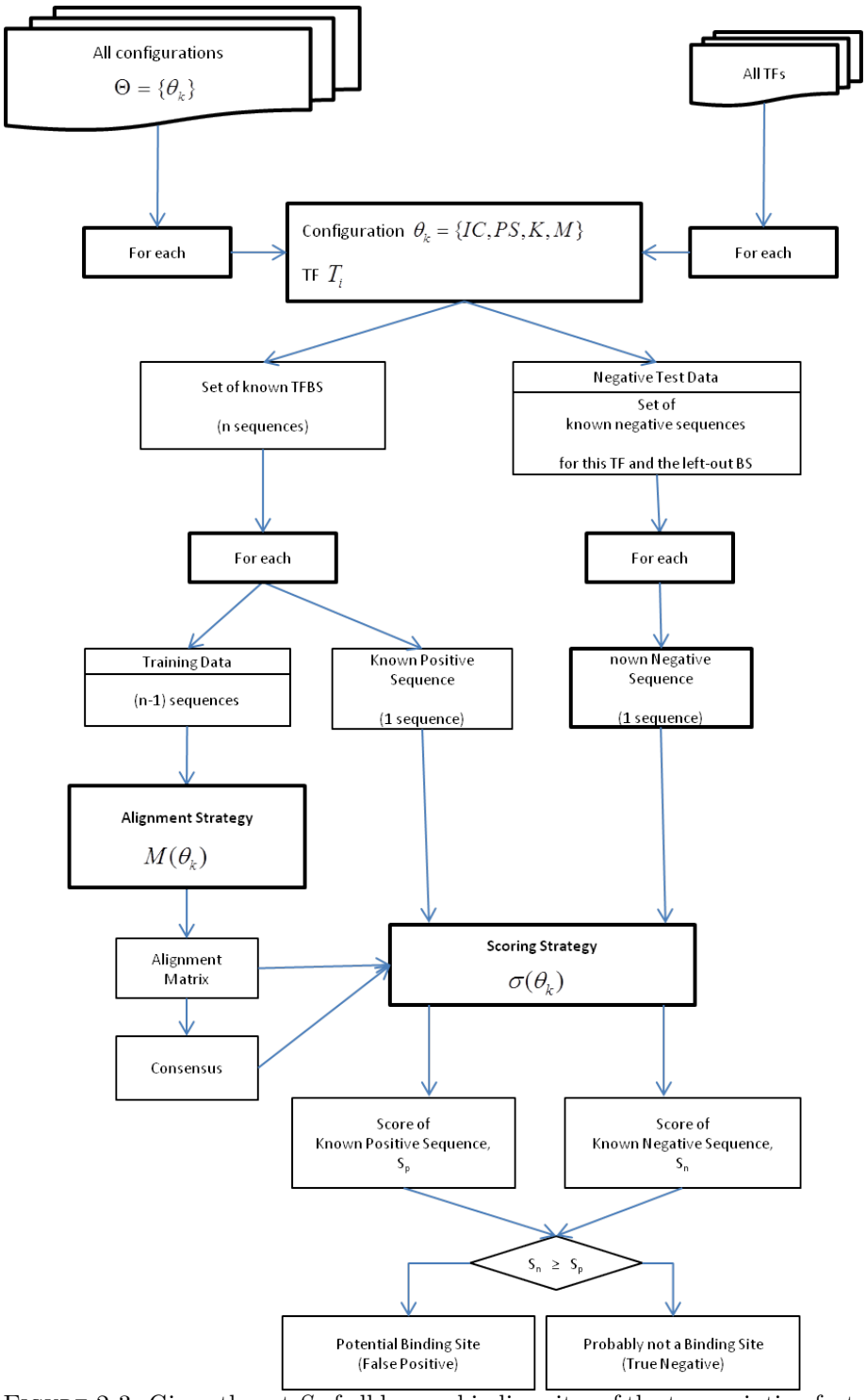

FIGURE 2.3: Given the set S of all known binding sites of the transcription factor T, a leave-one-out experiment is conducted for each binding site $p \in S$. In each of these experiments every known negative sequence t is scored in the ML-Consensus model. If its score is greater than or equal to the score of p, t is considered as a False Positive.

TF	BS index	Consensus length	Size of test dataset	Rank
dm001	0	10	341	0
dm001	1	8	341	0
dm001	2	6	341	23
dm001	3	7	341	2
dm001	4	8	341	1
dm001	5	7	341	3
dm001	6	8	341	0
dm001	7	8	341	0
dm001	8	7	341	35
dm001	9	11	341	0
dm001	10	7	341	5

Table 2.7: Outcome of the leave-one-out experiments for the configuration $\{IC = \texttt{False}, PS = \texttt{False}, K = \texttt{Null}, M = \texttt{Clustal}\}$ on the transcription factor `dm001`. Since the TF had 11 known binding sites, there were 11 leave-one-out experiments. The output file for each configuration contained entries for all 457 TFs. The test dataset for each leave-one-out experiment had 341 sequences (1 positive and 340 negative examples).The 'rank' column is the number of known negative examples which scored at least as high as the known positive example in each leave-one-out experiment. This statistic was used in Wilcoxon matched-pair signed ranks test for comparing the performance of two configurations. The rank is also the number of false positives which was used in computing the ROC curve for each configuration.

2.6 Statistical Tools

We used ROC curves and Wilcoxon matched pair signed-ranks test to compare the performance of different configurations.

Wilcoxon matched-pair signed-ranks test

It is a non-parametric test which can be used to compare the outcomes of two experiments A and B and verify whether these datasets differ significantly from each other [42, 43]. A and B must have the same number of data points (say n), where the A_i and B_i are outcomes of the two experiments at trial $1 \leq i \leq n$. Each data point $A_i \in A$ (similarly, each $B_i \in B$) must be independent, and the underlying distribution of the data points may be unknown. A data point A_i is called the *rank* of the trial i in experiment A.

There are n pairs $P = \{(A_i, B_i)\}, 1 \leq i \leq n$. For each pair $p \in P$, its contribution in the *rank-sum* of the experiments A and B are computed as follows. Let $d_i^A =$

$A_i - B_i$ and $d_i^B = B_i - A_i, 1 \leq i \leq n$ be the signed differences in favor of A and B (respectively) at trial i. The rank-sum of A, denoted by R_A, is the sum of all d_i^A such that $d_i^A > 0$. Similarly, rank-sum of B, denoted by R_B, is the sum of all d_i^B such that $d_i^B > 0$. Clearly, $R_A, R_B > 0$. Intuitively, if $abs(R_A - R_B)$ is high, it implies that the outcomes of the experiments A and B differ significantly. Which one did significantly better can be known from the signs of R_A and R_B. Formally, the Wilcoxon statistic Z was computed as follows.

$$Z = \left| \frac{\min(R_A, R_B) - 0.5 - n(n+1)/4}{\sqrt{n(n+1)(2n+1)/24}} \right|, \tag{2.21}$$

where $n = |A| = |B|$.

According to the one-tailed normal distribution, the thresholds for p-values 0.01 and 0.05 are $x_{0.01} = 2.33$ and $x_{0.05} = 1.65$, respectively. Significance of difference at p-value 0.01 was computed as follows. if $Z \geq x_{0.01}$, the difference in outcomes of A and B is significant with p-value 0.01. If $R_A > R_B$ (and a higher rank means better performance), this difference is in favor of A; otherwise, if $R_A < R_B$ (and a higher rank means better performance), the difference is in favor of B. In our implementation, we used the number of false positives as the rank of a leave-one-out experiment (that is, an individual data point) for a configuration (see Table 2.7). Therefore, a higher rank implied worse performance. Let the function *significant difference* be defined as follows:

$$sig(\theta_1, \theta_2) = \begin{cases} p & \text{If } \theta_1 \text{ is significantly better than } \theta_2 \text{ with } p\text{-value } p \\ -p & \text{If } \theta_2 \text{ is significantly better than } \theta_1 \text{ with } p\text{-value } p \\ 0 & \text{Otherwise} \end{cases}. \tag{2.22}$$

If $sig(\theta_i, \theta_j) > 0$, it means the configuration θ_i is significantly better than the configuration θ_j. The statement "configuration θ_i is *superior* to configuration θ_j" is equivalent to the above statement.

Receiver Operating Characteristic (ROC) Curves

ROC curves for each experiment configuration $\theta_k \in \Theta, 1 \leq k \leq N_{\text{config}}$ were produced from the outcome of each leave-one-out experiment. Let $\sigma_{\theta_k}(.)$ (σ in short) be the scoring function for the configuration θ_k. Let X_{Neg} be the set of all known negative examples for this experiment. Let p be the known positive sequence. Let $X_{\text{Neg}}^{\text{High}}$ be the set of known negative examples which scored higher than the known positive example. That is,

$$X_{\text{Neg}}^{\text{Highscore}} = \{t \in X_{\text{Neg}} : \sigma(t) \geq \sigma(p)\}$$

Let $N_{\text{Neg}}^{\text{High}}$ be the number of such known negative sequences. This number appears at the "Rank" column of Table 2.7. Let TP, TN, FP and FN denote the number of true positives, true negatives, false positives, and false negatives, respectively.

False positive rate, or FPR, is defined as the fraction of incorrectly classified known negative examples. Similarly, `true positive rate`, or TPR, is defined as the fraction of correctly classified known positive examples.

If the known positive example is to be considered a true positive, the sequences $t \in X_{\text{Neg}}^{\text{High}}$ cannot be considered as negative examples. Thus $N_{\text{Neg}}^{\text{High}}$ known negative examples must be *tolerated* before p can be classified as true positive. For a fixed set X_{Neg}, the number $N_{\text{Neg}}^{\text{High}}$ corresponds to a certain *tolerated false positive rate* $\text{FPR}_{\text{Tolerated}}$ and vice versa: for fixed X_{Neg}, every value of $\text{FPR}_{\text{Tolerated}}$ corresponds to a certain number α such that p can be classified as true positive only when $N_{Neg}^{High} \leq \alpha$. We considered allowable false positive rate from 0% to 20%, that is, $0 \geq \text{FPR}_{\text{Tolerated}} \geq 0.2$. This range was discretized into several intervals. For each leave-one-out experiment over a given set of TFBS, the true positive rate corresponding to each of these intervals were computed. These values were used to generate an ROC curve for this configuration.

The area under the ROC curve, or AUC in short, is a measure of the discriminatory

power of the underlying binary classifier [44, 45]. It is equal to the probability that if the classifier would be able to correctly differentiate between a known positive and a known negative example. In general a configuration with higher AUC is more powerful than a configuration with lower AUC. However, it is valid only in FPR region where the two curves do not intersect [44]. Since the ROC curves produced from leave-one-out experiments often intersect within the specified FPR region, we cannot directly compare the performance of two configurations solely based on the AUC. Therefore, a comparison based on AUC will be used only to complement a comparison based on Wilcoxon matched-pair signed-ranks test. However, two groups of configurations were compared by the *sum* of the AUC of the member configurations since this is the same as taking the average of a number of ROC curves and then taking the AUC of the averaged curve.

Details of Computing ROC Curves

Let N_{TF} be the number of TFs for the given species. Let TF_i be the i^{th} TF. Let N_{BS}^i be the number of known binding sites for TF_i. A leave-one-out cross-validation is conducted for each of the N_{BS}^i binding sites. If a known negative example scores more than the known positive example, it is considered as a false positive.

We computed an ROC (Receiver Operating Characteristic) curve for each configuration over each species. FPR and TPR were placed along x-axis and y-axis, respectively, and the curve indicates the TPR obtained at different values for FPR. The computation for each configuration was done in three steps. At first, we computed TPR and FPR for each leave-one-out experiment involving a known binding site. Next, these values were averaged over all BSs for each TF. Lastly, these values were further averaged over all TFs for a given species.

Step One: Individual binding sites. Let $\mathrm{BS}_{j,i}$ be the j^{th} BS of TF_i. Let FPR_{max} be the maximum false positive rate considered for drawing the ROC curve. We used

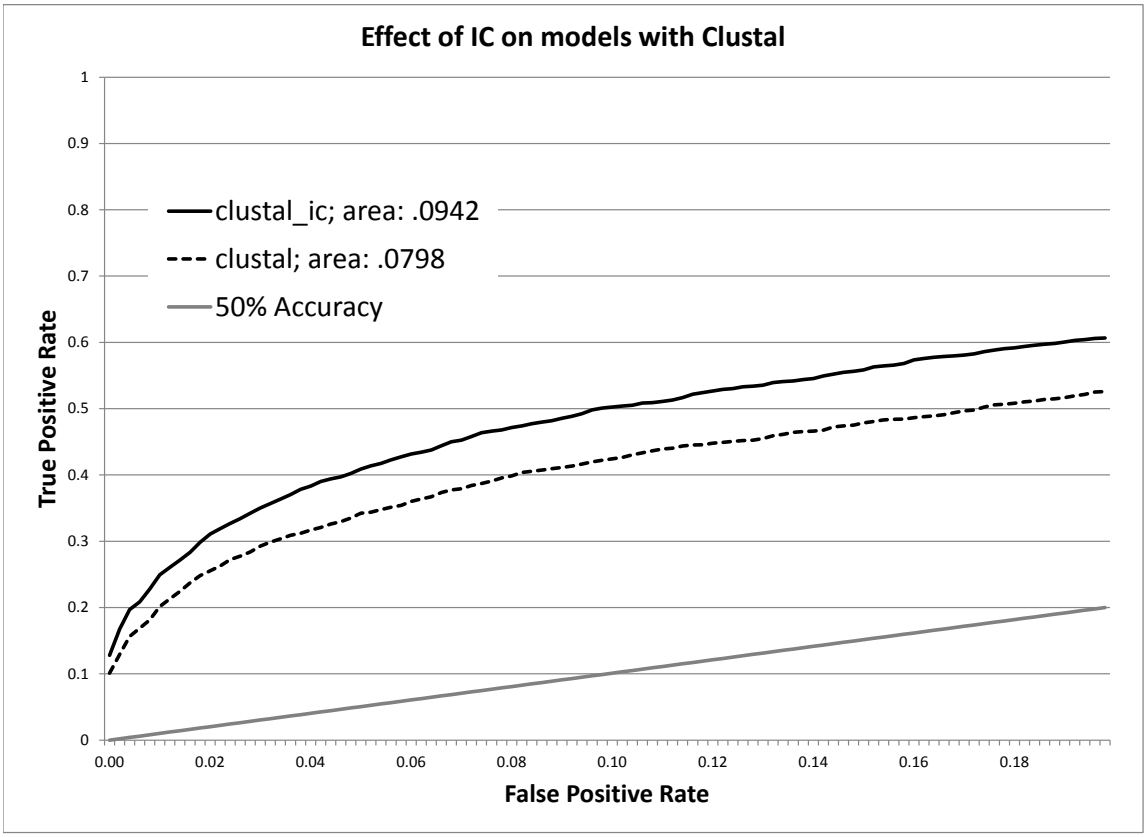

FIGURE 2.4: ROC curves for the configurations $\{IC = \texttt{True}, PS = \texttt{False}, K = \texttt{Null}, M = \texttt{Clustal}\}$ and $\{IC = \texttt{True}, PS = \texttt{False}, K = \texttt{Null}, M = \texttt{Clustal}\}$. The curve is drawn only within the FPR range of 0.0–0.2.

$\text{FPR}_{max} = 0.20$, or 20%. Let the range $0 \leq \text{FPR} \leq \text{FPR}_{max}$ be divided into M equal intervals. Let $\text{FPR}_k^{\text{interval}}$ denote the false positive rate corresponding to the k^{th} interval.

Let $\text{FP}_{j,i}$ be the number of false positives in the leave-one-out run which involves $\text{BS}_{j,i}$ as the known positive binding site. Let $\text{FPR}_{j,i}$ be the observed false positive rate. For any given *allowable* false positive rate, if $\text{FPR}_{j,i}$ is greater than the allowable FPR, the given configuration will not be able to identify the known positive example. $T_j(i,k)$ denotes whether the known positive example could be identified (i.e., occurrence of a true positive) by setting the allowable FPR equal to the false positive rate for the k^{th} FPR interval.

$$T_j(i,k) = \begin{cases} 1 & : \mathrm{FPR}_{j,i} \leq \mathrm{FPR}_k^{\mathrm{interval}} \\ 0 & : otherwise \end{cases} \quad , \tag{2.23}$$

for $1 \leq j \leq N_{\mathrm{BS}}^i,\ 1 \leq i \leq N_{\mathrm{TF}},\ 1 \leq k \leq M$.

Step Two: Averaging over all BSs for a given TF. For TF_i, let $T_{\mathrm{BS}}(i,k)$ be the average number of true positives obtained by setting the allowable FPR equal to the false positive rate for the k^{th} FPR interval.

$$T_{\mathrm{BS}}(i,k) = \frac{1}{N_{\mathrm{BS}}^i} \cdot \sum_{j=1}^{N_{\mathrm{BS}}^i} T_j(i,k) \tag{2.24}$$

for $1 \leq i \leq N_{\mathrm{TF}},\ 1 \leq k \leq M$.

Step Three: Averaging over all TFs. Let $T_{\mathrm{TF}}(k)$ be the average number of true positives obtained by setting the allowable FPR equal to the false positive rate for the k^{th} FPR interval across all TFs.

$$T_{\mathrm{TF}}(k) = \frac{1}{N_{\mathrm{TF}}} \cdot \sum_{i=1}^{N_{\mathrm{TF}}} T_{\mathrm{BS}}(i,k) \tag{2.25}$$

for $1 \leq k \leq M$. The ROC curve is produced by plotting $T_{\mathrm{TF}}(k)$ at k^{th} FPR interval.

We considered only 0%–20% false positive rate for computing the area under an ROC curve. Since the FPR intervals are discrete, we used the sum of TPR values in the mentioned FPR range as the area under an ROC curve. Figure 2.4

2.7 Analysis and Comparisons

As mentioned in Section 2.5, there were 144 different combinations of model parameters. For each of these combinations we conducted leave-one-out experiments over

the input dataset. There were two contexts for analyzing these experiments: (1) all experiments together (2) experiments grouped by species (there were 6 species). Thus each analysis was performed on all TFs together as well as on TFs from the same species grouped together.

Basic Analysis

ROC curves for each configuration. We computed ROC curves for each configuration $\theta \in \Theta$. Let $AUC(\theta)$ be the area under the ROC curve for configuration θ.

Pairwise Wilcoxon MPSR test. Next, we conducted Wilcoxon MPSR tests between every pair of configurations $\theta_i, \theta_j \in \Theta$. Let W be the $|\Theta| \times |\Theta|$ matrix where the $(i,j)^{\text{th}}$ element $w_{ij} = sig(\theta_i, \theta_j)$ (see Equation (2.22)).

We performed advanced analysis based on the two basic analyses above. These are the following.

Analysis 1. (AUC across all PS scopes)

Then, let $R(IC, M)$ be the set of for all configurations having the same IC and M. There are 12 possible combinations of the rest of the variables PS and K (namely no PS, PS scopes 1–10, and lastly full PS scope). Thus $|R(IC, M)| = 12$, for all combinations of the variables IC and M. Let $R(ic, m)$ be the set of configurations corresponding to a particular combination of IC and M. Then we plotted $AUC(\theta), \theta \in R(ic, m)$ with increasing PS scope (starting from no PS) which showed how the performance of a model (with fixed IC and M) changes with the change in PS scope. Let $\theta^* \in R(ic, m)$ be the configuration having the largest AUC. Then, its PS scope is called the *peak in AUC* for $R(ic, m)$.

Analysis 2. (Superiority of configurations with successive PS scopes)

Let $R(ic, m) = \{\theta_i\}, 1 \leq i \leq 12$. It can be seen that when $i \notin \{1, 12\}$, the PS scope of θ_i is $i-1$. The special case $i = 1$ denotes no PS, and $i = 12$ denotes full PS scope. For each configuration pair $(\theta_i, \theta_{i-1}), 2 \leq i \leq 12$ we examined if $sig(\theta_i, \theta_{i-1}) > 0$. That is, we examined the following: when IC and alignment strategy remained fixed, whether the performance of a configuration significantly increased with an increase in its PS scope.

Analysis 3. (Superiority between two groups of configurations)

Let $\pi \in \{IC, PS, K, M\}$ be a model parameter of ML-Consensus. For the configuration $\theta \in \Theta$, let $\theta(\pi)$ be the value of the parameter π in θ. Let Θ_A and Θ_B be two disjoint subsets of Θ such that (1) all configurations in each subset have the same value for the parameter π, (2) if two configurations $\theta_A \in \Theta_A$ and $\theta_B \in \Theta_B$ have the same values for all model parameters except π, then they must have different values for π. It follows that $|\Theta_A| = |\Theta_B| = n$. Let $p_i = (\theta_i^A \in \Theta_A, \theta_i^B \in \Theta_B)$ be the i^{th} pair of configurations. As per the property (2) above, the configurations of p_i have different value for parameter π but same values for all other parameters. Now let $Q_A = \{q_i\}, 1 \leq i \leq n$ be a sequence of values such that $q_i = 1 - sig(p_i(1), p_i(2))$. In effect, $q_i \in \{0, \pm 0.95, \pm 0.95\}$, and the sequence Q_A tells *how much* significantly better is each configuration θ_i^A over θ_i^B. Let the sequence $Q_B = \{-q_i\}, 1 \leq i \leq n$ be the negation of Q_A. Now, if a pair of configurations from (Θ_A,Θ_B) non-significant difference in performance, the sequences Q_A and Q_B will contain 0 in corresponding entries.

We constructed the partitions Θ_A and Θ_B based on the model parameters IC (two partitions) and M (six partitions). For alignment, we compared the partition corresponding to the naïve alignment with all other partitions. For each partition-pair, we constructed the sequences Q_A and Q_B, and conducted Wilcoxon MPSR test

on them with the following null hypothesis

$$H_0 : \theta_{M=\text{Naive}} \text{ cannot be significantly better than } \theta_{M\neq\text{Naive}} ,$$

the alternate hypothesis being

$$H_1 : \theta_{M=\text{Naive}} \text{ is significantly better than } \theta_{M\neq\text{Naive}} .$$

This comparison revealed whether configurations from a particular partition were *consistently* superior to their corresponding configurations from another partition.

3

Results

In this section, the phrase *performance of a configuration across different PS scopes* actually refers to the performance of all configurations with the same IC and M, while other parameters being

$$(PS, K) \in \{(\mathit{False},\ \mathit{Null}) \cup \{(\mathit{True}, k), 1 \leq k \leq 10\} \cup (\mathit{True}, \inf)\}$$

The configurations in this set are ordered by the PS scope value, that is, from no PS, to PS scopes 1–10, to full PS scope.

1. AUC, as a function of PS scope, showed global maxima.

When the PS scope of the model was varied from no PS, to PS scopes 1–10, to full PS scope, the AUC started decreasing after attaining its maximum value at certain scope. For a configuration with any combination of IC and M, when we plotted its AUC at different PS scopes (in ascending order), the resultant curve was bell-shaped.

2. Improvement in performance, as a function of PS scope, reached a plateau.

When only the PS scope of the model was varied from no PS, to PS scopes 1–10, to full PS scope, other parameters being the same, certain contiguous region of PS

scopes yielded significantly better performance than the PS scopes before or after the region. When the performance of a configuration at a certain PS scope was compared to the similar configuration at the previous PS scope, and the statistical significance of the difference of performance was measured, every combination of model parameters IC and M (with K varying across all PS scopes, including no PS) had a region of PS scopes where the performance increased significantly with increase in PS scope (the *upward* region), followed by a region where the performance did not vary significantly with increase in PS scope (the *significance plateau*), followed by a region where the performance decreased significantly with increase in PS scope (the *downward* region). With very few exceptions, there was only one significance plateau for each combination of parameters IC and M.

3. Configurations with naïve alignment yielded significantly better performance.

The difference between the performances of a configuration θ_{naive} with naïve alignment strategy and another configuration θ_{other} with some other alignment strategy (other parameters remaining the same) were, in most cases, either statistically significant in favor of θ_{naive}, or not statistically significant. There were only few cases where θ_{other} performed significantly better than θ_{naive}.

4. Configurations with IC yielded significantly better performance

Configurations using IC performed significantly better than configurations without using IC, other model parameters being the same. The improvement was more prominent at large PS scopes.

4

Discussion

4.1 PS scopes and Significance Plateau

From Equation (2.12) it can be observed that pairwise score at any PS scope K accounts for matches from all scopes $s < K$, plus new matches at scope K. In other words, score at a larger scope has all position-pair information gathered in all smaller scopes. (Table 2.3 demonstrates this fact.) Thus it was expected that a scoring function of a configuration using larger PS scope would be more powerful than that of a configuration with a smaller PS scope (other parameters being the same). Since AUC is a measure of how good a classifier is, the AUC of a configuration across all PS scopes (other parameters staying the same) was expected to be strictly non-decreasing. Similarly, a configuration with larger PS scope was expected to do significantly better (or at least no worse) than a configuration with smaller PS scope, other parameters being the same.

However, it was found (Result 3) that for all configurations, as PS scope increased, the AUC increased to a certain value and then decreased. The *AUC peak*, that is, the PS scope where the largest AUC occurred, was not the same for all configurations. Figure 4.1 demonstrates this behavior for configurations without IC.

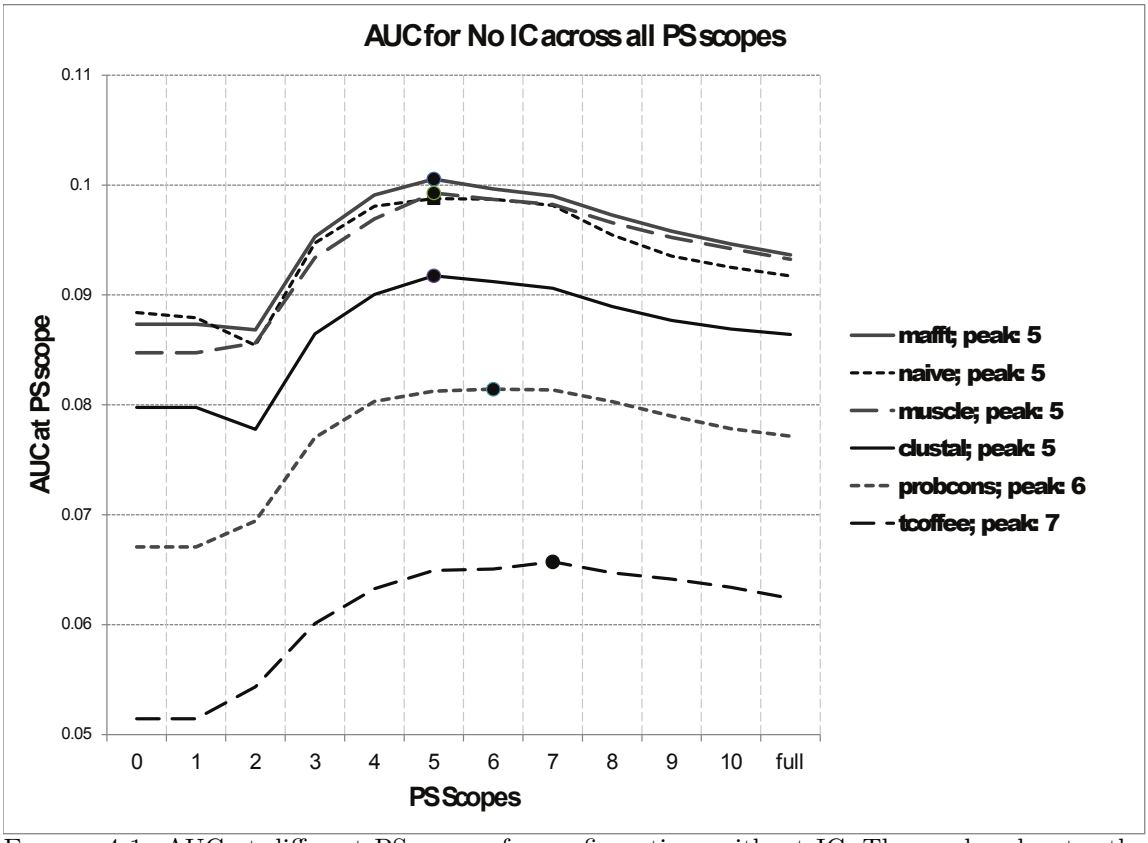

FIGURE 4.1: AUC at different PS scopes for configurations without IC. The marker denotes the location of the largest AUC.

Similarly, the existence of a significance plateau (Result 3) indicated that the performance of configurations (with the same IC and alignment strategy) were significantly better at a specific region of PS scope values than at neighboring PS scopes. Moreover, it also indicated that the performance did not improve (nor deteriorated) within the plateau. Note that it is possible that the upward region was directly followed by the downward region, without any level region in between. This situation simply indicated that after a significant increase in performance at scope k, the performance significantly deteriorated at scope $k + 1$.

Explanation

These observations can be explained as follows. For any leave-one-out experiment over a given set of TFBSs, the known positive example (if not identical to the consensus) had one or more mismatches with respect to the consensus. Some of these positions were involved in position-pair matches between the consensus and a known negative example. Let us call such an event *noise*. A noise event increased the probability that the known negative example would score higher than the known positive example — producing a false positive. PS scopes, when larger than a certain value, did not capture any new position-pair matches, yet continued picking up noise. This is why we observed a decrease in performance of a configuration with increase in PS scopes beyond this value.

Since a consensus generalizes the similarity of a set of binding sites, the PS scope that yielded the maximum AUC may indicate information about the interdependence of positions within the core region of the binding sites. However, how this information can be characterized or used is not clear at this point, and is a matter of future work.

Table 4.1 shows the significance plateaus and locations of AUC peaks for different configurations over the entire input dataset. Figure 4.2 shows the significance plateau and the AUC peaks on the same curve. From Table 4.1 it can be seen that

1. The significance plateau was present in every combination of IC and alignment strategy.
2. Length of the plateau was either 1 or 2.
3. In 9 out of the 12 cases, the significance plateau started at scope 4.
4. In 11 out of the 12 cases, the scope 4 was inside the significance plateau.
5. The scope yielding the maximum AUC fell outside the significance plateau.

	Without IC			With IC		
Config	Successive Improvement	Plateau	AUC Peak	Successive Improvement	Plateau	AUC Peak
Clustal	=..//_.....	4 - 4	5	=..//=.....	4 - 5	5
Naïve	=..//_.....	4 - 4	5	=../=/_.==...	5 - 5	8
MAFFT	=..//_.....	4 - 4	5	=..//=.....	4 - 5	5
ProbCons	=../=.....	3 - 4	6	=..//=.....	4 - 5	7
Muscle	=..//_.....	4 - 4	5	=../=.....	4 - 5	7
T-Coffee	=../=.....	3 - 4	7	=..//=.....	4 - 5	7

Table 4.1: Statistical significance of change of performance of configurations with an increment in PS scope. The column "Successive Improvement" shows the sequence of changes in performance. The characters '/','=', and '.' refer to significant increase, significant decrease, and non-significant change in performance, respectively. The significance plateau is marked with underline. The column "Plateau" marks the start and the end of the plateau. The column "AUC Peak" refers to the PS scope that yielded largest AUC.

Interpretation

For any configuration, the beginning of a significance plateau is the PS scope after which no significant improvement in performance occurred. Moreover, this scope indicates the maximum distance within which pairwise positional dependencies occurred in an aligned set of TFBSs. However, it should be noted that the value of this scope (as found in our study) was a function of the alignment strategy and scoring strategy (that is, IC) in use.

Since the PS scope 4 was found to be present in every significance plateau, and since it was the beginning of the plateau in 9 out of 12 cases (see Table 4.1), we can make the following generalized claim.

Claim *4 is the distance which most of the pairwise positional dependencies occurred in the core region of a known set of TFBSs.*

Caveat: This claim has only empirical basis, and was not substantiated with any biological or theoretical basis from our work. Moreover, this value, as we arrived

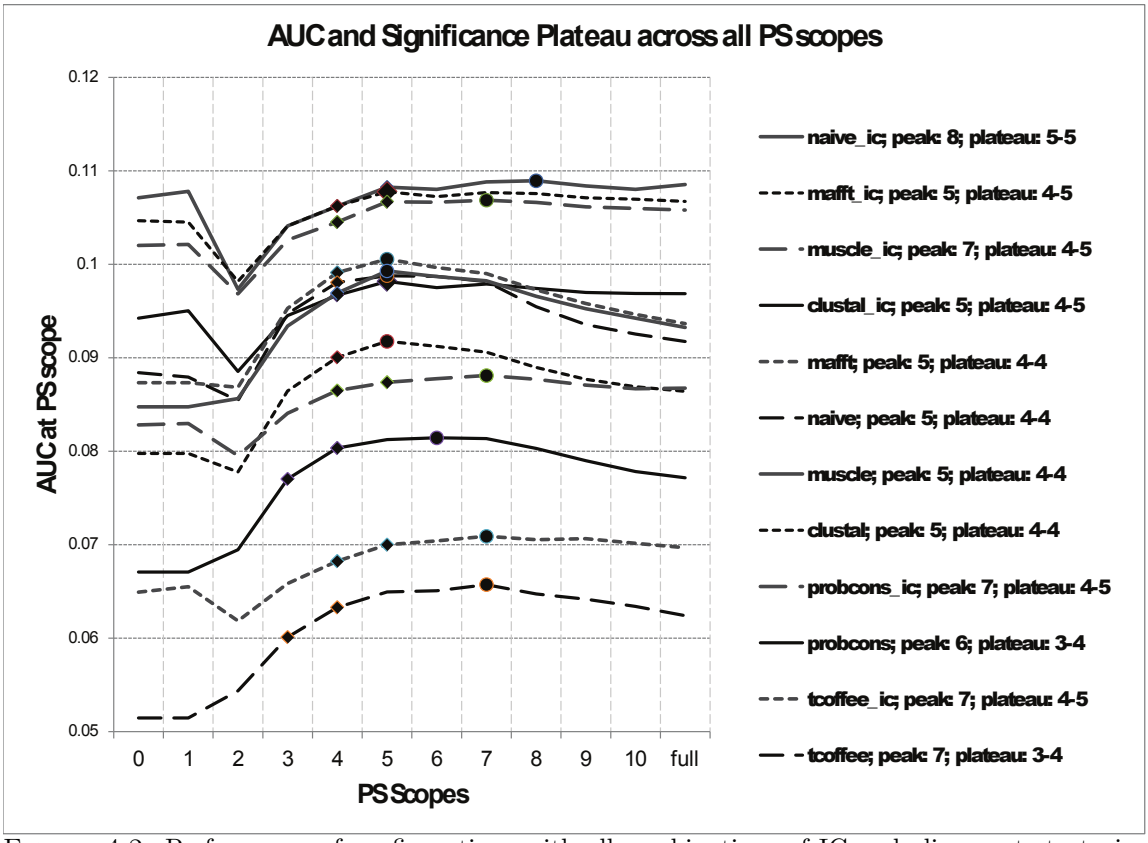

FIGURE 4.2: Performance of configurations with all combinations of IC and alignment strategies across all PS scopes. Small diamond markers denote the significance plateau. A round marker denotes the AUC peak. A large diamond marker indicates that AUC peak lies inside the significance plateau. The chart legends show configuration names (along with its AUC peak), from top to bottom, sorted by the sum of AUC across all PS scopes, from high to low.

at it, is a function of alignment strategy and scoring function (that is, usage of information content). Additionally, there is an underlying assumption that a core region is shared by these TFBSs.

Additionally, we can assume that the core region should be at least as large as the distance within which most pairwise positional dependencies *actually* occur in an aligned set of TFBSs. Therefore, we have the following claim.

Claim *The length of the core region from a set of TFBSs is at least 4.*

Caveat: This claim has only empirical basis, and was not substantiated with any biological or theoretical basis from our work. Moreover, this value, as we arrived at it, is a function of alignment strategy and scoring function (that is, usage of

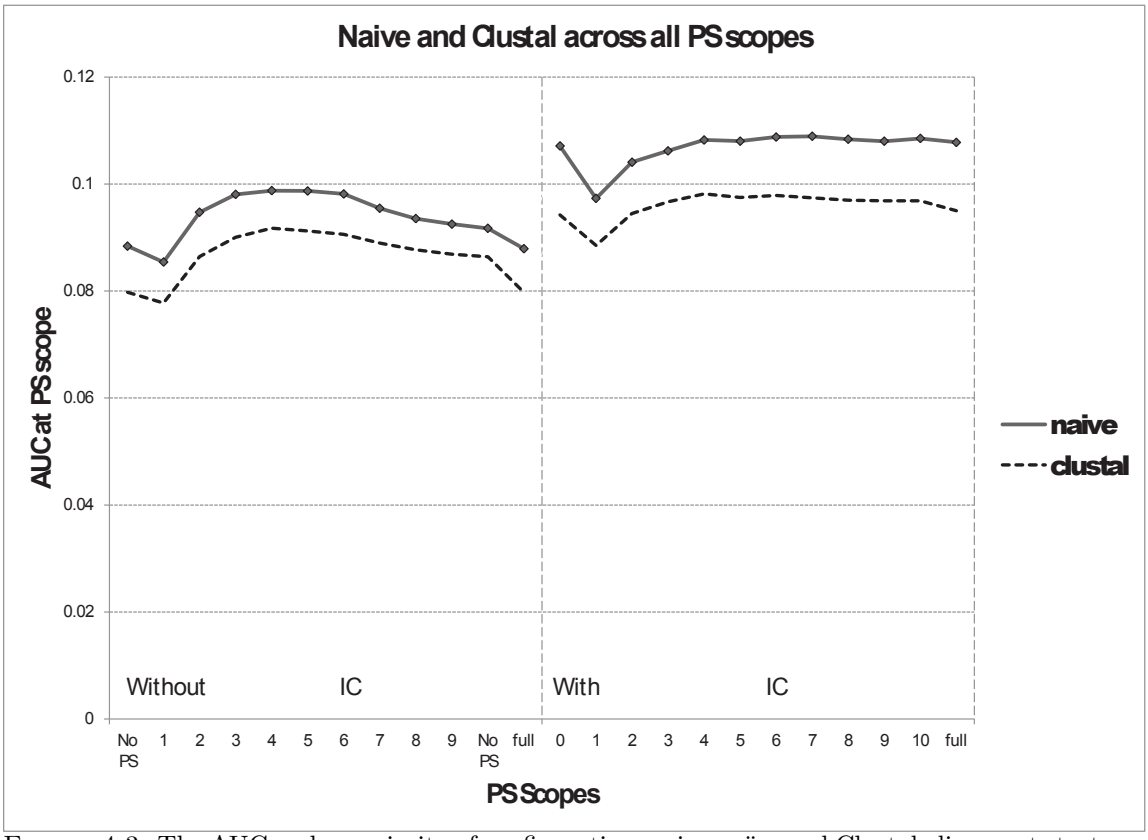

FIGURE 4.3: The AUC and superiority of configurations using naïve and Clustal alignment strategy. The diamond markers denote where the improvement in performance is statistically significant.

information content).

4.2 Comparing Naïve with Other Alignment Strategies.

The results from Analysis 1 (AUC areas at different PS scopes) and Analysis 2 (Superiority of configurations with successive PS scopes), when placed together for configurations with two different alignment strategies, it was consistently found that the configurations with naïve alignment strategy were superior to the configurations with another alignment strategy. Figures 4.3, 4.8, 4.5, 4.6 and 4.7 depict the comparisons mentioned above.

Moreover, Tables 4.2 and 4.3 show that configurations using naïve alignment strategy were *almost always* found superior to configurations using other alignment strate-

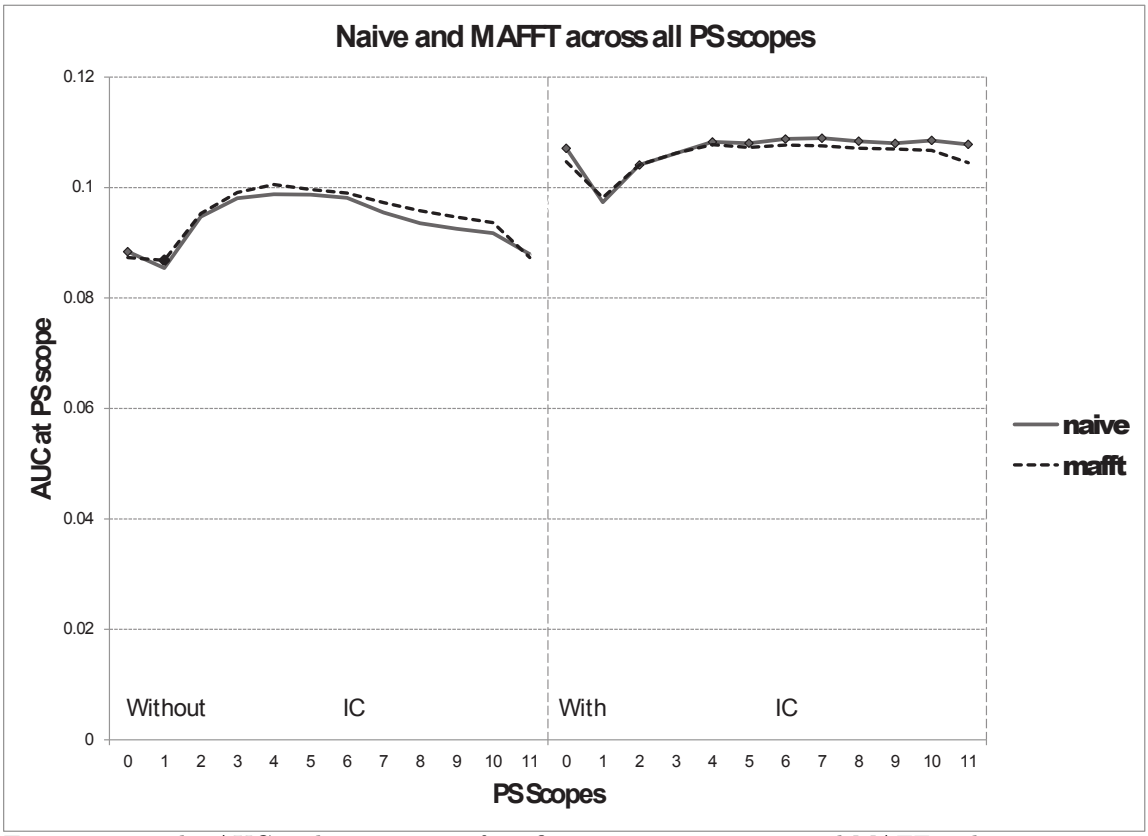

FIGURE 4.4: The AUC and superiority of configurations using naïve and MAFFT alignment strategy. The diamond markers denote where the improvement in performance is statistically significant.

gies. When all leave-one-out experiments were considered, configurations using naïve alignment was always superior. When experiments were grouped by species, there was only 6 out of 30 cases where configurations using naïve were not superior.

Next, we partitioned the set of all configurations by alignment strategy, and compared the partition $\Theta_{M=\text{Naive}}$ with other partitions $\Theta_{M\neq\text{Naive}}$ using Analysis 2.7 (see Section 2.7). We made the following null hypothesis:

$$H_0 : \theta_{M=\text{Naive}} \text{ cannot be significantly better than } \theta_{M\neq\text{Naive}} ,$$

the alternate hypothesis being

$$H_1 : \theta_{M=\text{Naive}} \text{ is significantly better than } \theta_{M\neq\text{Naive}} .$$

The results are presented in Table 4.4, which shows that the said results were significantly in favor of the configurations using naïve alignment strategy.

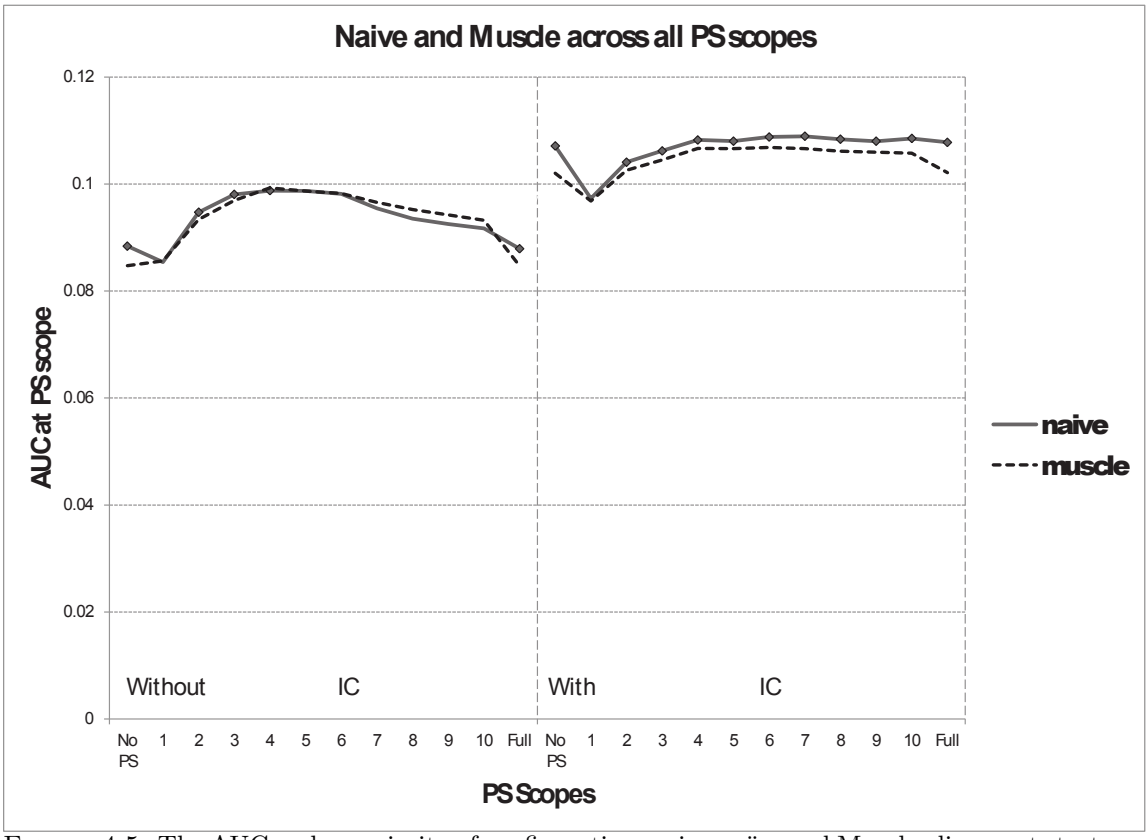

FIGURE 4.5: The AUC and superiority of configurations using naïve and Muscle alignment strategy. The diamond markers denote where the improvement in performance is statistically significant.

Explanation

The naïve alignment strategy operated on simple assumptions and it did not do anything as sophisticated as other multiple sequence alignment strategies. However, it used information content and pairwise dependence information in its alignment, which improved the quality of the alignment, measured by the performance of respective configurations. Another possibility is that the number of input sequences was small (˜ 10 in each set) and the variability was high, which *may have* worked against one or more assumptions of these tools about the input context.

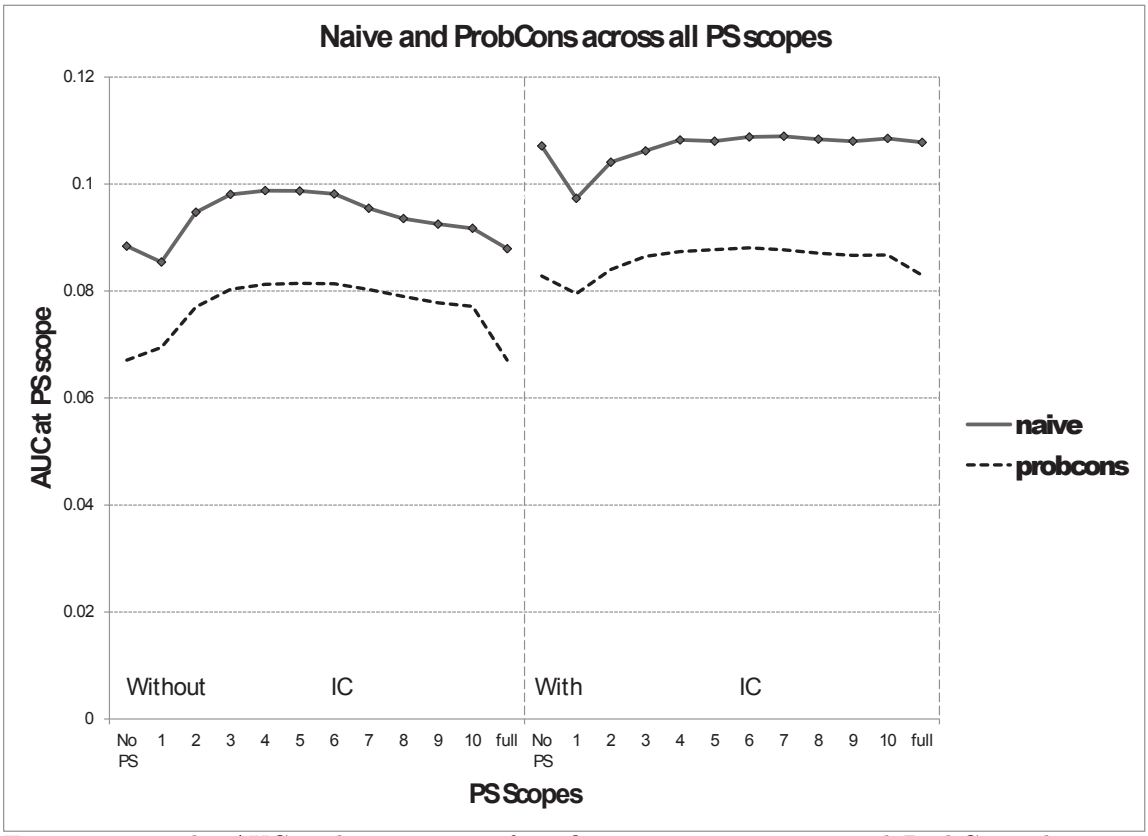

FIGURE 4.6: The AUC and superiority of configurations using naïve and ProbCons alignment strategy. The diamond markers denote where the improvement in performance is statistically significant.

Interpretation

Since the performance of configurations using the naïve algorithm were significantly better than the performance of configurations using another alignment strategy, we can make several conclusions.

Conjecture *These state-of-the-art algorithms may not, as yet, be capable of producing the optimal multiple sequence alignment for a given set of known TFBSs.*

It can be argued that the input parameters (that is, command line options) for each alignment tool could have been fine-tuned so that the performance for each tool would be maximized. But it should also be noted that the default options for each tool are optimized for *good enough* performance on general input data. Additionally, these alignment tools were capable of making insertions/deletions, which gave them

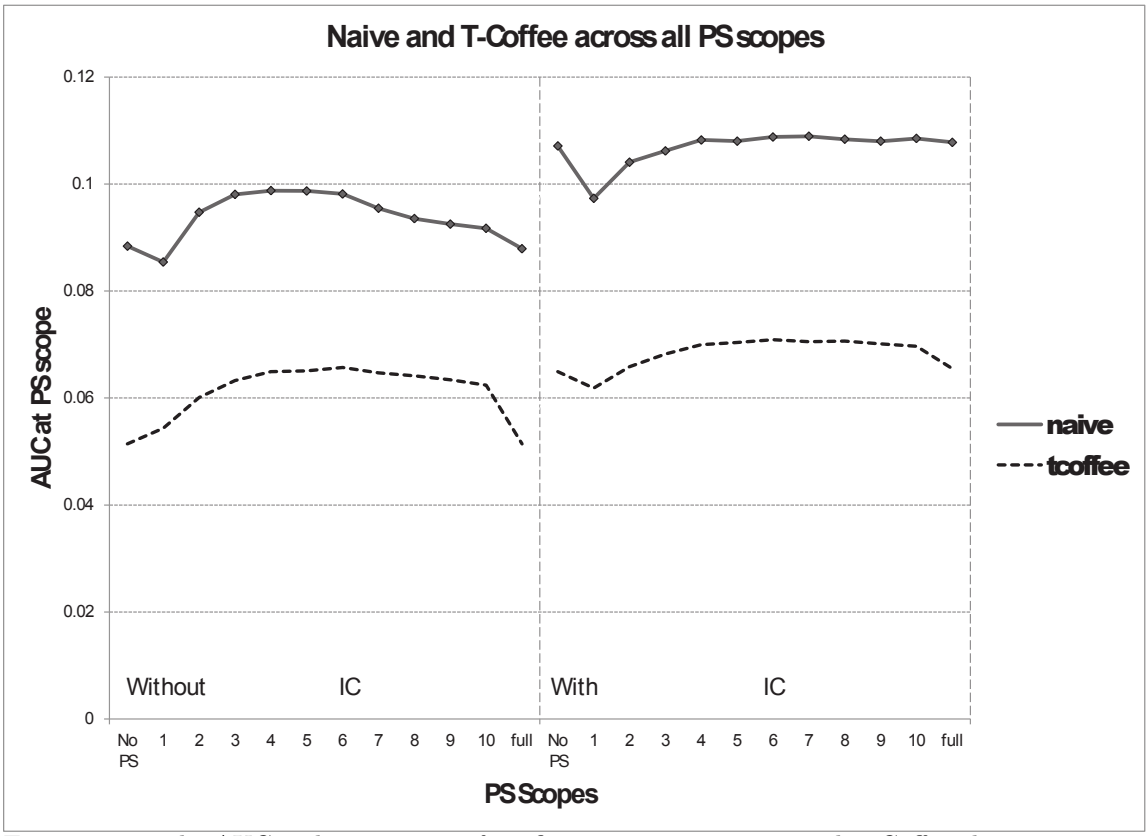

FIGURE 4.7: The AUC and superiority of configurations using naïve and T-Coffee alignment strategy. The diamond markers denote where the improvement in performance is statistically significant.

advantage in making a better alignment. However, it *may* also happen that the core region for a set of TFBSs does not involve gaps, which is why the presence of gaps within the core region led to decreased performance of the configurations that used that consensus.

The alignment tools used in our study are all general-purpose, which means they were not designed with TFBS alignment in mind. If such information and assumptions could be incorporated into their algorithms, it may have improved their performance for aligning TFBSs.

Suggestion. *A new multiple sequence alignment tool can be designed specifically for aligning TFBSs. This tool will use all state-of-the-art techniques along with prior information and assumptions about TFBSs.*

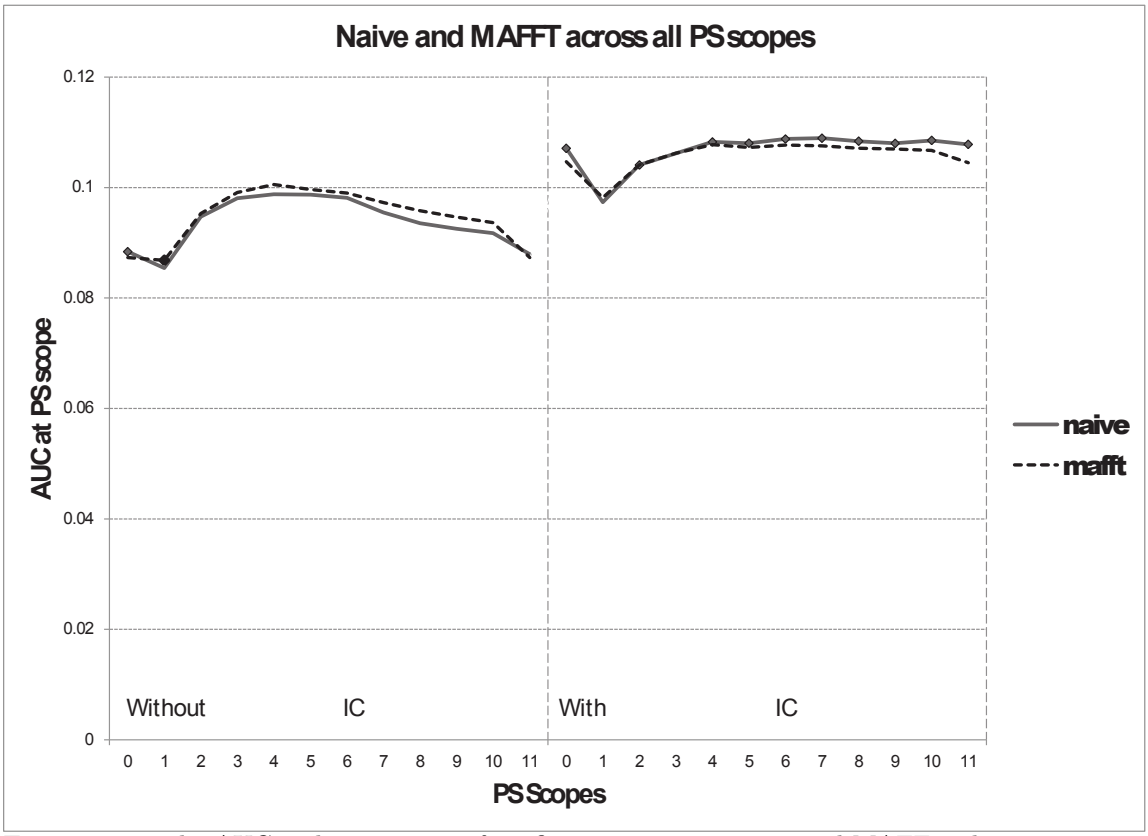

FIGURE 4.8: The AUC and superiority of configurations using naïve and MAFFT alignment strategy. The diamond markers denote where the improvement in performance is statistically significant. Without using IC, the AUC of configurations with naïve alignment strategy were usually less than the AUC of configurations with MAFFT. However, no alignment strategy led to consistently superior performance. But when using IC, configurations with naïve alignment were found to have larger AUC than, and perform consistently superior to, the configurations with MAFFT alignment.

For example, an assumption made by the naïve alignment algorithm was that there is a core region contained by all binding sites. Examples of prior information about TFBS are the minimum length of the core region (Claim 2 in Section 4.1) and maximum distance of pairwise dependence in the core region (Claim 1 in Section 4.1). However, it is not clear at this point how these information could be used in designing such an algorithm. This is a matter of future work.

Outline of the proposed TFBS alignment algorithm

This algorithm will use iteratively align the given set of TFBS. At each step it will compute some estimate α about the core region from the current alignment. An

Experiment Group	Comparing naïve against	Superior Alignment	Wilcoxon Z
All	Clustal	Naïve	269.1886
dm	Clustal	Naïve	71.1014125
gg	Clustal	Naïve	48.75143774
hm	Clustal	Naïve	183.3972924
mus	Clustal	Naïve	128.4772227
rat	Clustal	Naïve	108.6515612
yst	Clustal	Clustal	56.18188707
All	MAFFT	Naïve	245.3510539
dm	MAFFT	MAFFT	62.42723972
gg	MAFFT	MAFFT	42.59227624
hm	MAFFT	Naïve	169.0611192
mus	MAFFT	Naïve	116.4019818
rat	MAFFT	Naïve	98.4713736
yst	MAFFT	MAFFT	52.54208235
All	Muscle	Naïve	259.1131893
dm	Muscle	Naïve	69.07588605
gg	Muscle	Muscle	44.22487953
hm	Muscle	Naïve	177.5697366
mus	Muscle	Naïve	124.7208924
rat	Muscle	Naïve	100.872844
yst	Muscle	Muscle	55.95663949

Table 4.2: Comparison between configurations with naïve against configurations with Clustal, MAFFT, and Muscle alignment strategies in all configurations. The value 'All' in the first column (Experiment Group) means all experiments were compared, whereas other values (dm, gg, hm, mus, rat, yst) mean only experiments corresponding to TFs of a certain species were compared. The second column is the alignment strategy being compared to naïve strategy. The third column (Superior Alignment) denotes the superior configuration with p-value 0.01. The fourth column (Wilcoxon Z) is the Wilcoxon statistic for the comparison. Note that threshold statistic Z at p-value 0.01 is 2.33.

example of such a metric is the beginning of the significance plateau. Additionally, it will compute the fitness (using an appropriate function) of the current alignment. As long as the fitness is below some threshold, it will compute the alignment for the next step utilizing α from current step. The algorithm will stop when the fitness will be above the threshold or when a local maxima of fitness will be achieved.

Experiment Group	Comparing Naïve against	Superior Alignment	Wilcoxon Z
All	ProbCons	Naïve	289.0121799
dm	ProbCons	Naïve	73.39235116
gg	ProbCons	Naïve	52.00832555
hm	ProbCons	Naïve	197.8888903
mus	ProbCons	Naïve	136.7944862
rat	ProbCons	Naïve	114.0762977
yst	ProbCons	Naïve	67.63078512
All	T-Coffee	Naïve	299.310314
dm	T-Coffee	Naïve	75.63217741
gg	T-Coffee	Naïve	53.16137995
hm	T-Coffee	Naïve	204.4663442
mus	T-Coffee	Naïve	142.0964926
rat	T-Coffee	Naïve	116.5373824
yst	T-Coffee	Naïve	74.35983621

Table 4.3: Comparison between configurations with naïve against configurations with Probcons and T-Coffee alignment strategies in all configurations. The value 'All' in the first column (Experiment Group) means all experiments were compared, whereas other values (dm, gg, hm, mus, rat, yst) mean only experiments corresponding to TFs of a certain species were compared. The second column is the alignment strategy being compared to naïve strategy. The third column (Superior Alignment) denotes the superior configuration with p-value 0.01. The fourth column (Wilcoxon Z) is the Wilcoxon statistic for the comparison. Note that threshold statistic $Z_{\text{critical},p=0.01} = 2.33$.

4.3 Effect of Information Content

Configurations that used information content in scoring function (and also in alignment when naïve algorithm was used) performed significantly better than the configurations that did not use it (Figure 4.9). It can be observed from Figure 4.9 that the rate at which the AUC decreased after attaining the maximum value is smaller when IC was used in configurations compared to when IC was not used in configurations. In other words, configurations having larger PS scopes benefited more (compared to configurations using smaller PS scopes) when both used IC.

Additionally, it can be seen from Table 4.1 that when IC was used, the significance plateau was either elongated by 1 scope, or was shifted by 1 scope towards larger values, compared to the significance plateau when IC was not used.

The positive impact of IC in the model matches with our expectation, as the same

Input Subset	% cases H_0 Holds	Z	p-value	Superior
H_0 : Naïve not significantly better than Clustal				
All	0	4.3	0.01	Naïve
dm	0	4.3	0.01	Naïve
gg	0	4.3	0.01	Naïve
hm	0	4.3	0.01	Naïve
mus	0	4.3	0.01	Naïve
rat	0	4.3	0.01	Naïve
yst	29.16	0.549125	—	—
H_0 : Naïve not significantly better than MAFFT				
All	4.16	2.5887	0.01	Naïve
dm	0	0	—	—
gg	4.16	2.1325	0.05	Naïve
hm	0	2.9785	0.01	Naïve
mus	4.16	1.9604	0.05	Naïve
rat	0	3.5421	0.01	Naïve
yst	45.83	1.0888	—	—
H_0 : Naïve not significantly better than Muscle				
All	0	3.5421	0.01	Naïve
dm	0	2	0.05	—
gg	0	2.0083	0.05	Naïve
hm	0	4.1231	0.01	Naïve
mus	8.33	0.5394	—	Naïve
rat	0	1.7889	0.05	Naïve
yst	8.33	0.5394	—	—

Input Subset	% cases H_0 Holds	Z	p-value	Superior
H_0 : Naïve not significantly better than ProbCons				
All	0	4.3	0.01	Naïve
dm	0	4.3	0.01	Naïve
gg	0	4.3	0.01	Naïve
hm	0	4.3	0.01	Naïve
mus	0	4.3	0.01	Naïve
rat	0	4.3	0.01	Naïve
yst	0	4.3	0.01	Naïve
H_0 : Naïve not significantly better than T-Coffee				
All	0	4.3	0.01	Naïve
dm	0	4.3	0.01	Naïve
gg	0	4.3	0.01	Naïve
hm	0	4.3	0.01	Naïve
mus	0	4.3	0.01	Naïve
rat	0	4.3	0.01	Naïve
yst	0	4.3	0.01	Naïve

Table 4.4: Statistical significance of the difference in the results of pairwise Wilcoxon MPSR test between configurations with naïve and other alignment strategies, all other parameters remaining the same.

was observed in [31] for fixed-length datasets. These above findings imply that the incorporation of IC in the scoring function should be a strongly desirable aspect of a TFBSs recognition model using consensus.

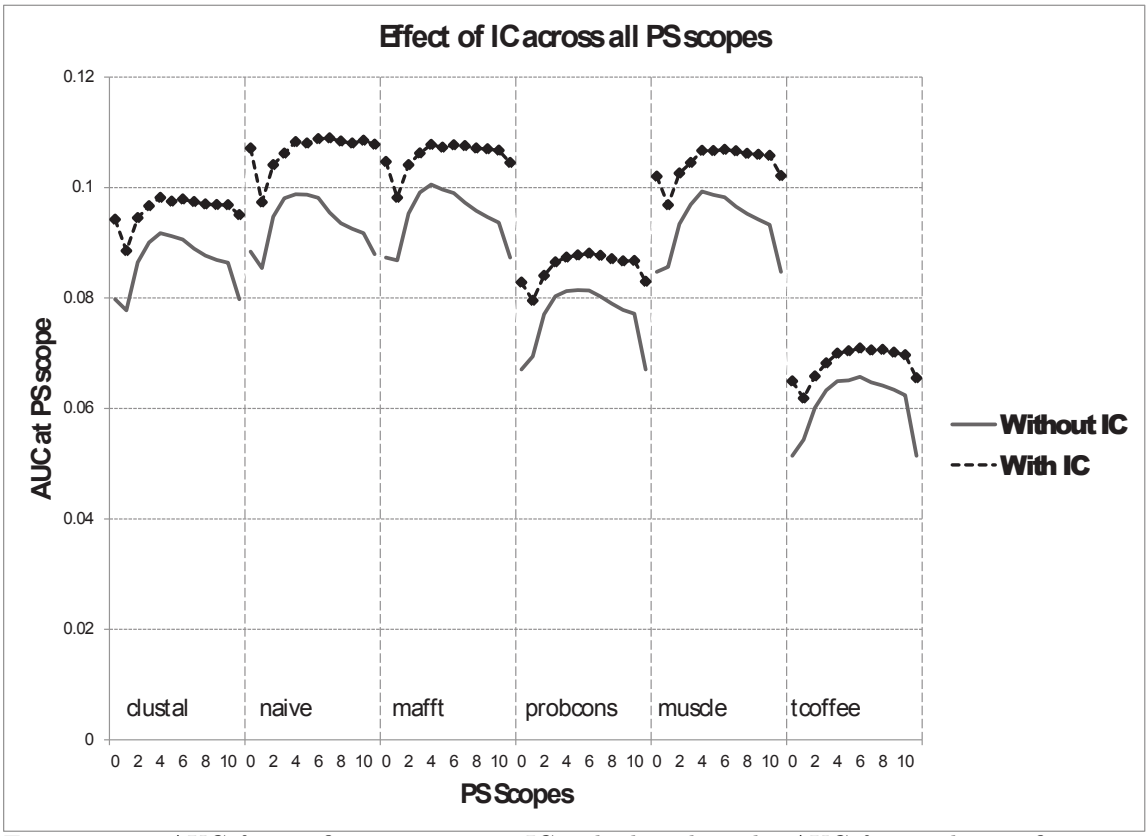

FIGURE 4.9: AUC for configurations using IC is higher than the AUC for similar configuration without IC. In every case, a configuration with IC performed significantly better than a configuration without IC.

4.4 Extensions to the ML-Consensus Model

The ML-Consensus model can be extended in several ways. The first, and most obvious, way is to use a PWM in place of a consensus as a representation model. A PWM also contains the IC with it which will greatly improve the scoring function of the model. Secondly, any alignment strategy (for example, expectation-maximization algorithms) with some desirable property can be used with the model as long as it generates an alignment matrix. Thirdly, a cut-off value can be computed with the model so that it can be used to globally classify an input sequence based on this threshold. Fourth, currently the model can classify one short sequence at a time. Thus the input mechanism could be altered so that it would accept long DNA strings

and identify all potential binding sites within that DNA string.

Bibliography

[1] Gary D Stormo. DNA binding sites: representation and discovery. *Bioinformatics*, 16(1):16–23, 2000.

[2] G Z Hertz, G W Hartzell, and G D Stormo. Identification of consensus patterns in unaligned DNA sequences known to be functionally related. *Computer applications in the biosciences : CABIOS*, 6(2):81–92, April 1990.

[3] William H E Day and F R McMorris. Critical comparison of consensus methods for molecular sequences. *Nucleic Acids Research*, 20(5):1093–1099, 1992.

[4] Thomas D Schneider. Consensus sequence Zen. *Applied bioinformatics*, 1(3):111–9, January 2002.

[5] G D Stormo and D S Fields. Specificity, Free Energy and Information Content in Protein-DNA Interactions. *Trends in Biochemical Sciences*, 23:109–113, 1998.

[6] T D Schneider and R M Stephens. Sequence logos: a new way to display consensus sequences. *Nucleic acids research*, 18(20):6097–100, October 1990.

[7] S. Altschul. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, September 1997.

[8] K Cartharius, K Frech, K Grote, B Klocke, M Haltmeier, A Klingenhoff, M Frisch, M Bayerlein, and T Werner. MatInspector and beyond: promoter analysis based on transcription factor binding sites. *Bioinformatics*, 21(13):2933–2942, 2005.

[9] T L Bailey and C Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proceedings International Conference on Intelligent Systems for Molecular Biology ISMB International Conference on Intelligent Systems for Molecular Biology*, 2(6):28–36, 1994.

[10] Timothy L Bailey, Nadya Williams, Chris Misleh, and Wilfred W Li. MEME: discovering and analyzing DNA and protein sequence motifs. *Nucleic Acids Research.*, 34 (Web Se:W369–W373, 2006.

[11] F P Roth, J D Hughes, P W Estep, and G M Church. Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. *Nature biotechnology*, 16(10):939–45, October 1998.

[12] D S Chekmenev, C Haid, and A E Kel. P-Match: transcription factor binding site search by combining patterns and weight matrices. *Nucleic Acids Research*, 33(Web Server issue):W432–W437, 2005.

[13] Albin Sandelin, Wynand Alkema, P\"{a}r Engstr\"{o}m, Wyeth W Wasserman, and Boris Lenhard. JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucl. Acids Res.*, 32 (Supple:D91–D94, 2004.

[14] C T Workman and G D Stormo. ANN-Spec: a method for discovering transcription factor binding sites with improved specificity. In *Pacific Symposium on Biocomputing 5*, pages 464–475, 2000.

[15] Falcon F M Liu, Jeffrey J P Tsai, R M Chen, S N Chen, and S H Shi. FMGA: Finding Motifs by Genetic Algorithm. *Fourth IEEE Symposium on Bioinformatics and Bioengineering (BIBE'04)*, bibe:459, 2004.

[16] Nuno D Mendes, Ana C Casimiro, Pedro M Santos, Isabel Sá-Correia, Arlindo L Oliveira, and Ana T Freitas. MUSA: a parameter free algorithm for the identification of biologically significant motifs. *Bioinformatics*, 22(24):2996–3002, 2006.

[17] Chen Yanover, Mona Singh, and Elena Zaslavsky. M are better than one: an ensemble-based motif finder and its application to regulatory element prediction. *Bioinformatics*, 25(7):868–874, 2009.

[18] Pavel A Pevzner and Sing-Hoi Sze. Combinatorial approaches to finding subtle signals in DNA sequences. In *ISMB*, pages 269–278. AAAI Press, 2000.

[19] Eugen Fazius, Vladimir Shelest, and Ekaterina Shelest. SiTaR: a novel tool for transcription factor binding site prediction. *Bioinformatics (Oxford, England)*, 27(20):2806–2811, September 2011.

[20] Modan K Das and Ho-Kwok Dai. A survey of DNA motif finding algorithms. *BMC Bioinformatics*, 8(Suppl 7), 2007.

[21] K Robison, A M McGuire, and G M Church. E. coli DNA-Binding Site Matrices Applied to the Complete E. coli K12 Genome. *J. Mol. Biol.*, 284:241–254, 1998.

[22] T Riley, E Sontag, P Chen, and A Levine. Transcriptional control of human p53-regulated genes. *Nat Rev Mol Cell Biol*, 9(5):402–412., 2008.

[23] E Soldaini, S John, S Moro, J Bollenbacher, U Schindler, and W J Leonard. DNA binding site selection of dimeric and tetrameric Stat5 proteins reveals a large repertoire of divergent tetrameric Stat5a binding sites. *Mol Cell Biol*, 20:389–401, 2000.

[24] G B Ehret, P Reichenbach, U Schindler, C M Horvath, S Fritz, M Nabholz, and P Bucher. DNA binding specificity of different STAT proteins. Comparison of in vitro specificity with natural target sites. *J Biol Chem*, 276(9):6675–6688, 2001.

[25] John E Reid, Kenneth J Evans, Nigel Dyer, Lorenz Wernisch, and Sascha Ott. Variable structure motifs for transcription factor binding sites. *BMC genomics*, 11(30):30, January 2010.

[26] Cédric Notredame. Recent Evolutions of Multiple Sequence Alignment Algorithms. *PLoS Computational Biology*, 3(8):4, 2007.

[27] G Badis and Others. Diversity and Complexity in DNA Recognition by Transcription Factors. *Science*, 324:1720–1723, 2009.

[28] Martha L Bulyk, Philip L F Johnson, and George M Church. Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors. *Nucleic Acids Research*, 30(5):1255–1261, 2002.

[29] Yoseph Barash, Gal Elidan, Nir Friedman, and Tommy Kaplan. Modeling dependencies in protein-DNA binding sites. In *Proceedings of the seventh annual international conference on Computational molecular biology - RECOMB '03*, pages 28–37, New York, New York, USA, April 2003. ACM Press.

[30] Qing Zhou and Jun S Liu. Modeling within-motif dependence for transcription factor binding site predictions. *Bioinformatics (Oxford, England)*, 20(6):909–16, April 2004.

[31] Robert Osada, Elena Zaslavsky, and Mona Singh. Comparative analysis of methods for representing and searching for transcription factor binding sites. *Bioinformatics*, 20(18):3516–3525, December 2004.

[32] E Wingender, X Chen, R Hehl, H Karas, I Liebich, V Matys, T Meinhardt, M Pruss, I Reuter, and F Schacherer. TRANSFAC: an integrated system for gene expression regulation. *Nucl. Acids Res.*, 28(1):316–319, 2000.

[33] Saad Quader, Nathan Snyder, Kevin Su, Ericka Mochan, and Chun-Hsi Huang. ML-Consensus: a general consensus model for variable-length transcription factor binding sites. In Clara Pizzuti, Marylyn D. Ritchie, and Mario Giacobini, editors, *EvoBIO'11 Proceedings of the 9th European conference on Evolutionary computation, machine learning and data mining in bioinformatics*, pages 25–36, Rome, April 2011. Springer-Verlag Berlin, Heidelberg.

[34] Saad Quader and Chun-Hsi Huang. Effect of Positional Dependence and Alignment Strategy on Modeling Transcription Factor Binding Sites. *BMC research notes*, 5(1):340, July 2012.

[35] A Cornish-Bowden. IUPAC-IUB Symbols for Nucleotide Nomenclature. *Nucl. Acids Res*, 13:3021–3030, 1985.

[36] Thomas D Schneider, Gary D Stormo, and Larry Gold. Information Content of Binding Sites on Nucleotide Sequences. *J. Mol. Biol.*, 188:415–431, 1986.

[37] Fabian Sievers, Andreas Wilm, David Dineen, Toby J Gibson, Kevin Karplus, Weizhong Li, Rodrigo Lopez, Hamish McWilliam, Michael Remmert, Johannes Söding, Julie D Thompson, and Desmond G Higgins. Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. *Molecular systems biology*, 7(539):539, January 2011.

[38] Kazutaka Katoh, Kazuharu Misawa, Kei-ichi Kuma, and Takashi Miyata. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic acids research*, 30(14):3059–3066, 2002.

[39] Robert C Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic acids research*, 32(5):1792–1797, 2004.

[40] Chuong B Do, Mahathi S P Mahabhashyam, Michael Brudno, and Serafim Batzoglou. ProbCons: Probabilistic consistency-based multiple sequence alignment. *Genome research*, 15(2):330–340, 2005.

[41] C Notredame, D G Higgins, and J Heringa. T-Coffee: A novel method for fast and accurate multiple sequence alignment. *Journal of molecular biology*, 302(1):205–17, September 2000.

[42] David J Sheskin. *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC, 2000.

[43] The Wilcoxon Matched-Pairs Signed-Ranks Test.

[44] Paolo Sonego, András Kocsor, and Sándor Pongor. ROC analysis: applications to the classification of biological sequences and 3D structures. *Briefings in bioinformatics*, 9(3):198–209, May 2008.

[45] David L Streiner and John Cairney. What's under the ROC? An introduction to receiver operating characteristics curves. *Canadian journal of psychiatry. Revue canadienne de psychiatrie*, 52(2):121–8, February 2007.